



---

# **ExpressLane PEX 8619-BA**

## **16-Lane, 16-Port PCI Express Gen 2**

### **Switch with DMA**

### **Data Book**

Version 1.3

**March 2011**

Website [wwwplxtech.com](http://wwwplxtech.com)

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

Phone 800 759-3735

408 774-9060

FAX 408 774-2169

**PLX CONFIDENTIAL - NDA REQUIRED**

---

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – Version 1.3  
March, 2011

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

## Revision History

Version	Date	Description of Changes
1.0	July, 2009	Production Release, Silicon Revision BA.
1.1	May, 2010	Production update, Silicon Revision BA. Added support for Industrial temperature, and added new order part number.
1.3	March, 2011	Production update, Silicon Revision BA. Applied miscellaneous corrections and enhancements throughout the data book.

## Copyright Information

Copyright © 2009 – 2011 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 Dual Cast, 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 National Semiconductor Corporation.

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

Document Number: 8619-BA-SIL-DB-P1-1.3

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – **PLX CONFIDENTIAL - NDA REQUIRED**

## 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 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA, for hardware designers and software/firmware engineers.

## Supplemental Documentation

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

- PLX Technology, Inc. (PLX), [wwwplxtech.com](http://wwwplxtech.com)
  - The [PLX PEX 8619 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, [www.standardics.nxp.com](http://www.standardics.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 to PCI Bridge Architecture Specification, Revision 1.2*
  - *PCI Express Base Specification, Revision 1.0a*
  - *PCI Express Base Specification, Revision 1.1*
  - *PCI Express Base Specification, Revision 2.0*
  - *PCI Express Base Specification, Revision 2.0 Errata*
  - *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](#)
- Personal Computer Memory Card International Association (PCMCIA), [www.pcmcia.org](http://www.pcmcia.org)
  - *ExpressCard Standard Release 1.0*
- PXI Systems 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*

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – **PLX CONFIDENTIAL - NDA REQUIRED**

**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 r1.0a</i>	<i>PCI Express Base Specification, Revision 1.0a</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 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – **PLX CONFIDENTIAL - NDA REQUIRED**

## Terms and Abbreviations

The following table lists common terms and abbreviations used in this data book. Terms and abbreviations defined in the *PCI Express Base r2.0* are 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).
ACK	Acknowledge Control Packet. Control packet used by a destination to acknowledge data packet receipt. Signal that acknowledges signal receipt.
AMCAM	Address-mapping CAM that determines a memory Request route. Contains mirror copies of the PCI-to-PCI bridge <b>Memory Base</b> and <b>Memory Limit</b> registers in the switch.
ARI	Alternative Routing-ID Interpretation.
BAR	Base Address register.
BER	Bit error rate.
BusNoCAM	Bus Number-mapping CAM that determines the Completion route. Contains mirror copies of the PCI-to-PCI bridge <b>Secondary Bus Number</b> and <b>Subordinate Bus Number</b> registers in the switch.
CAM	Content-Addressable Memory.
CDR	Clock Data Recovery.
CRC	Cyclic Redundancy Check.
CSRs	Configuration Space registers.
Data MRd	Data Memory Read. Reading from the Source address, to gather data to transfer.
Data MWr	Data Memory Write. Writing to the Destination address, to transfer data that was gathered with a Read.
Descriptor	16 bytes of information that describes the DMA, <i>such as</i> the Source address to read, Destination address to write, and total Transfer size.
DLL	Data Link Layer.
DMA	Direct Memory Access.
DMA Ring	Physical location where the DMA Descriptors reside. When the last Descriptor is done, the pointer automatically moves to the location of the first Descriptor, creating a DMA Descriptor Ring.
Downstream Device	Device that is connected to a downstream Port.
Downstream Port	Port that is used to communicate with a device below it within the system hierarchy. A switch can have one or more downstream Ports.
ECC	Error-Correcting Code.
ECRC	End-to-end Cyclic Redundancy Check.
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.
GPIO	General-Purpose Input/Output.
GT/s	Giga-Transfers per second.
Half Station	Internal hardware processing partition of Even- or Odd-numbered (Even or Odd) Ports.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

Terms and Abbreviations	Definitions
INCH	Ingress Credit Handler.
InitFC	Initialization Flow Control.
IOCAM	I/O Address-mapping CAM that determines an I/O Request route. Contains mirror copies of the PCI-to-PCI bridge <b>I/O Base</b> and <b>I/O Limit</b> registers in the switch.
JTAG	Joint Test Action Group.
Lane	A bidirectional pair of differential PCI Express I/O signals.
LCRC	Link Cyclic Redundancy Check.
Link Interface	Primary side of the NT Port, connects to external device pins. The secondary side of the NT Port is called 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 credits</i> ) that belong to the PCI Express Link logic.
LTSSM	Link Training and Status State Machine.
LUT	Lookup Table.
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.
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 8619 supports one NT Port.
Partial Completion	In PCI Express, a single Read can have one or more Completions with data. If more than one Completion is returned, they are referred to as <i>Partial Completions</i> .
PCI Express Link Logic	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.
PEC	Packet Error Code.
PEX	PCI Express.
PHY	Physical Layer.
PIPE	PHY Interface for PCI Express architecture.
PLL	Phase-Locked Loop.
PM	Power Management.
PME	Power Management Event.
Port	Interface to a group of SerDes and supporting logic that is capable of creating a Link, for communication with another 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Terms and Abbreviations	Definitions
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.
SPI	Serial Peripheral Interface.
SSC	Spread-Spectrum Clock.
Sticky Bits	Register bits in which the current values are unchanged by a Hot Reset, Link Down event, or Secondary Bus Reset, while the switch is powered. Sticky bits are reset to default values by a Fundamental Reset. Applies to ROS, RW1CS, and RWS CSR types, and sometimes HwInit. (Refer to <a href="#">Table 14-5, “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.
TC	Traffic Class.
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.
TS1	Type 1 Training Sequence Ordered-Set.
TS2	Type 2 Training Sequence Ordered-Set.
Tx	Transceiver.
UI	Unit Interval – 400 ps at 2.5 GT/s, 200 ps at 5.0 GT/s.
Upstream Device	Device that is connected to the upstream Port.
Upstream Port	Port that is used to communicate with a device above it within the system hierarchy.
UTP	User Test Pattern.
VC	Virtual Channel. The PEX 8619 supports two Virtual Channels, VC0 and VC1.
VCO	Voltage-Controlled Oscillator.
Vector	Address and data.
Virtual Interface	Secondary side of the NT Port, connects to the internal virtual PCI Express interface.
WRR	Weighted Round-Robin scheduling.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## Data Book Notations and Conventions

Notation / Convention	Description
Blue text	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	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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – **PLX CONFIDENTIAL - NDA REQUIRED**

# 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 16-Lane, 16-Port Switch	5
2.1.1	Highly Flexible Port Configurations	5
2.1.2	Non-Blocking Crossbar Switch Architecture	5
2.1.3	Low Packet Latency and High Performance	6
2.1.3.1	Data Payloads	6
2.1.3.2	Cut-Thru	6
2.1.4	Virtual Channels and Traffic Classes	7
2.1.5	Port Arbitration	7
2.1.6	End-to-End Packet Integrity	7
2.1.7	Configuration Flexibility	7
2.1.8	Interoperability	8
2.1.9	Low Power with Granular SerDes Control	8
2.1.10	Dynamic Lane Reversal	8
2.1.11	<i>performancePAK</i>	9
2.1.11.1	Read Pacing	9
2.1.11.2	Dual Cast	9
2.1.11.3	Dynamic Buffer Pool	9
2.1.11.4	Integrated Direct Memory Access Engine	9
2.1.12	<i>visionPAK</i>	10
2.1.12.1	Performance Monitoring	10
2.1.12.2	Error Injection	10
2.1.12.3	SerDes Loopback	10
2.1.12.4	SerDes Eye Capture	10
2.1.13	Hot Plug for High Availability	11
2.1.14	Fully Compliant Power Management	11
2.1.15	General-Purpose Input/Output Signals	11
2.1.16	Non-Transparent Dual Host and Failover Support – NT Mode Only	11
2.1.17	Spread-Spectrum Clock Isolation	11
2.2	Applications	12
2.2.1	Fan-Out	12
2.2.2	Control Plane Application	13
2.2.2.1	DMA in Control Planes for Large Systems – DMA Mode Only	14
2.2.3	Distributed Computing	15
2.2.4	Intelligent Adapter Board Redundancy – NT Mode Only	16
2.2.5	Generic Data Mover	17
2.3	Software Usage Model	18
2.3.1	System Configuration	18
2.3.2	Interrupt Sources and Events	18

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Chapter 3</b>	<b>Signal Ball Description</b>	<b>19</b>
3.1	Introduction	19
3.2	Abbreviations	19
3.3	Internal Pull-Up Resistors	20
3.4	Signal Ball Descriptions	20
3.4.1	PCI Express Signals	21
3.4.2	Serial Hot Plug Signals	23
3.4.3	Serial EEPROM Signals	24
3.4.4	Strapping Signals	25
3.4.5	JTAG Interface Signals	31
3.4.6	I <sup>2</sup> C/SMBus Slave Interface Signals	32
3.4.7	Device-Specific Signals	34
3.4.8	External Resistor Signals	36
3.4.9	No Connect Signals	36
3.4.10	Power and Ground Signals	37
3.5	Physical Layout	38
<b>Chapter 4</b>	<b>Functional Overview</b>	<b>39</b>
4.1	Hardware Architecture	39
4.1.1	Port Functions	40
4.1.1.1	Port Configurations	40
4.1.1.2	Port Numbering	41
4.2	PCI Express Link Functional Description	42
4.3	Physical Layer	43
4.3.1	Physical Layer Features	43
4.3.2	Physical Layer Status and Command Registers	44
4.3.3	Hardware Link Interface Configuration	44
4.4	Transaction Layer	45
4.4.1	Locked Transactions	47
4.4.2	Relaxed Ordering	47
4.4.3	TL Transmit/Egress Protocol – End-to-End Cyclic Redundancy Check	47
4.4.4	TL Receive/Ingress Protocol	47
4.4.5	Flow Control Credit Initialization	48
4.4.6	Flow Control Protocol	48
4.5	PCI-Compatible Software Model	49
<b>Chapter 5</b>	<b>Reset and Initialization</b>	<b>51</b>
5.1	Reset	51
5.1.1	Fundamental Reset	52
5.1.2	Hot Reset	52
5.1.3	Secondary Bus Reset	52
5.1.4	Register Bits that Affect Hot Reset	53
5.1.5	Reset and Clock Initialization Timing	53
5.2	Initialization Procedure	54
5.2.1	Default Port Configuration	55
5.2.2	Default Register Initialization	55
5.2.3	Device-Specific Registers	55
5.2.4	Serial EEPROM Load Time	56
5.2.5	I <sup>2</sup> C Load Time	56

<b>Chapter 6</b>	<b>Serial EEPROM Controller . . . . .</b>	<b>57</b>
6.1	Overview . . . . .	57
6.2	Features . . . . .	58
6.3	Serial EEPROM Load following Upstream Port Reset . . . . .	58
6.4	Serial EEPROM Data Format . . . . .	59
6.5	Serial EEPROM Initialization . . . . .	62
6.6	PCI Express Configuration, Control, and Status Registers . . . . .	63
6.7	Serial EEPROM Registers . . . . .	63
6.8	Serial EEPROM Random Write/Read Access . . . . .	64
6.8.1	Writing to Serial EEPROM . . . . .	64
6.8.2	Reading from Serial EEPROM . . . . .	65
6.8.3	Programming a Blank Serial EEPROM . . . . .	65
6.9	Serial EEPROM Loading of NT Port Link Interface Registers – NT Mode Only . . . . .	66
6.10	NT Port Expansion ROM – NT Mode Only . . . . .	66
<b>Chapter 7</b>	<b>I<sup>2</sup>C/SMBus Slave Interface Operation . . . . .</b>	<b>67</b>
7.1	Introduction . . . . .	67
7.2	I <sup>2</sup> C Slave Interface . . . . .	67
7.2.1	I <sup>2</sup> C Support Overview . . . . .	67
7.2.2	I <sup>2</sup> C Addressing – Slave Mode Access . . . . .	69
7.2.3	I <sup>2</sup> C Slave Interface Register . . . . .	69
7.2.4	I <sup>2</sup> C Command Format . . . . .	70
7.2.5	I <sup>2</sup> C Register Write Access . . . . .	71
7.2.5.1	I <sup>2</sup> C Register Write . . . . .	74
7.2.6	I <sup>2</sup> C Register Read Access . . . . .	76
7.2.6.1	I <sup>2</sup> C Register Read Address Phase and Command Packet . . . . .	79
7.2.6.2	I <sup>2</sup> C Register Read Data Packet . . . . .	80
7.3	SMBus Slave Interface . . . . .	81
7.3.1	SMBus Features . . . . .	81
7.3.2	SMBus Operation . . . . .	82
7.3.3	SMBus Commands Supported . . . . .	82
7.3.3.1	SMBus Block Write . . . . .	83
7.3.3.2	SMBus Block Read . . . . .	87
7.3.3.3	CSR Read, Using SMBus Block Read - Block Write Process Call . . . . .	91
7.3.4	SMBus Address Resolution Protocol . . . . .	92
7.3.4.1	SMBus UDID . . . . .	93
7.3.4.2	SMBus Supported ARP Commands . . . . .	95
7.3.5	SMBus PEC Handling . . . . .	97
7.3.6	Addressing PEX 8619 SMBus Slave . . . . .	97
7.3.7	SMBus Timeout . . . . .	98
7.4	Switching between SMBus and I <sup>2</sup> C Bus Protocols . . . . .	98
<b>Chapter 8</b>	<b>Virtual Channels and Port Arbitration . . . . .</b>	<b>99</b>
8.1	Quality of Service Support . . . . .	99
8.2	Virtual Channel Support . . . . .	99
8.2.1	Traffic Class to Virtual Channel Mapping . . . . .	99
8.2.2	Ingress VC Arbitration . . . . .	99
8.2.3	Egress VC Arbitration . . . . .	99
8.2.3.1	Fixed-Priority VC Arbitration . . . . .	100
8.2.3.2	Round-Robin VC Arbitration . . . . .	100
8.2.3.3	Weighted Round Robin Arbitration . . . . .	100

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

8.3 Queue Handling .....	101
8.4 VC1 Ingress Credit .....	102
8.4.1 VC1 .....	102
8.4.1.1 VC1 Threshold Registers .....	102
8.4.1.2 Programming the VC1 Global Enable and Threshold Registers .....	103
8.4.1.3 Usage Example .....	103
<b>Chapter 9 Performance Features.....</b>	<b>105</b>
9.1 Introduction .....	105
9.2 DLLP Policies .....	106
9.2.1 ACK DLLP Policy .....	106
9.2.2 UpdateFC DLLP Policy .....	108
9.2.3 Unidirectional DLLP Policies .....	110
9.3 Ingress Resources .....	111
9.3.1 Initial Credit Allocation .....	112
9.3.2 Dynamic Buffering .....	114
9.3.3 Ingress Credit Handler Threshold Registers (Offsets A00h through B7Ch) .....	117
9.3.4 Adjusting Initial Credit Values (Ingress Resources) .....	121
9.3.5 Credit Allocation When Common Pool Is Consumed .....	123
9.3.6 INCH Port Pool Registers (Offsets 940h through 94Ch) .....	125
9.3.7 Wait for ACK – Avoiding Congestion .....	126
9.4 Latency .....	127
9.4.1 Host-Centric Latency .....	128
9.4.2 Peer-to-Peer Latency .....	129
9.4.3 Other Latency Measurements .....	129
9.5 Queuing Options .....	130
9.5.1 Destination Queuing .....	131
9.5.2 Source Queuing .....	132
9.5.3 Port Arbitration .....	133
9.5.4 Port Bandwidth Allocation .....	133
9.6 Read Pacing .....	134
9.6.1 Read Pacing Example .....	134
9.6.2 Read Spacing (Spreading) Logic .....	136
9.6.3 Read Threshold .....	137
9.6.4 Read Pacing Benefits .....	137
9.6.5 Enabling Read Pacing and Read Spreading .....	138
9.7 Dual Cast .....	139
9.7.1 Introduction .....	139
9.7.2 Dual Cast System Model .....	139
9.7.3 Dual Cast Control Registers .....	141
9.7.3.1 Dual Cast Low BAR[0-7], Dual Cast High BAR[0-7] .....	141
9.7.3.2 Dual Cast Low BAR[0-7] Translation, Dual Cast High BAR[0-7] Translation .....	142
9.7.3.3 Dual Cast Low BAR[0-7] Setup, Dual Cast High BAR[0-7] Setup .....	143
9.7.3.4 Dual Cast Source Destination Port .....	145
9.7.4 Dual Cast Programming Example .....	146
9.7.4.1 Register Programming Steps .....	146
9.7.5 Dual Cast to a Non-Transparent Destination Port .....	146
9.7.6 Error Reporting of Failed Dual Cast Cycles .....	146

<b>Chapter 10</b>	<b>Integrated DMA Engine – DMA Mode Only . . . . .</b>	<b>147</b>
10.1	Features . . . . .	147
10.2	Terminology . . . . .	149
10.3	General Description . . . . .	150
10.4	DMA Usage Models . . . . .	151
10.4.1	Statistics Gathering . . . . .	151
10.4.2	Peer-to-Peer Transfer . . . . .	152
10.4.3	Move Data in Main Memory . . . . .	152
10.4.4	Endpoint . . . . .	152
10.4.5	Interprocessor Communication . . . . .	152
10.5	Software Interface . . . . .	153
10.6	DMA Descriptors . . . . .	154
10.6.1	Standard Descriptor Format . . . . .	155
10.6.2	Extended Descriptor Format . . . . .	156
10.6.3	DMA On-Chip Descriptor Mode . . . . .	157
10.6.4	DMA Off-Chip Descriptor Mode . . . . .	158
10.6.5	DMA Descriptor Writeback . . . . .	159
10.7	DMA Alignment . . . . .	159
10.8	Performance Considerations . . . . .	161
10.8.1	DMA Maximum Bandwidth . . . . .	161
10.8.2	DMA Maximum Round-Trip Read Time . . . . .	162
10.8.3	DMA and Non-DMA Policies . . . . .	163
10.8.3.1	DMA Traffic Class . . . . .	163
10.8.3.2	DMA Relaxed Ordering . . . . .	164
10.8.3.3	DMA No Snoop . . . . .	164
10.9	DMA Software Access . . . . .	165
10.9.1	Configure a DMA Transfer . . . . .	165
10.9.2	DMA Pause . . . . .	166
10.9.3	DMA Abort . . . . .	166
10.9.4	DMA Runtime Status . . . . .	167
10.9.5	DMA Routing Check . . . . .	167
10.9.6	DMA Error Header Logging . . . . .	167
10.9.7	DMA Current Status . . . . .	167
<b>Chapter 11</b>	<b>Interrupts. . . . .</b>	<b>169</b>
11.1	Interrupt Support . . . . .	169
11.1.1	Interrupt Sources or Events . . . . .	170
11.1.2	Interrupt Handling . . . . .	174
11.2	INTx Emulation Support . . . . .	175
11.2.1	INTx-Type Interrupt Message Re-Mapping and Collapsing . . . . .	176
11.2.1.1	Interrupt Re-Mapping and Collapsing – NT PCI-to-PCI Bridge Mode . . . . .	177
11.3	MSI Support . . . . .	178
11.3.1	MSI Operation . . . . .	178
11.3.1.1	NT PCI-to-PCI Bridge Mode MSIs – NT PCI-to-PCI Bridge Mode . . . . .	179
11.3.2	MSI Capability Registers . . . . .	179
11.4	PEX_INTA# Interrupts . . . . .	180
11.5	General-Purpose Input/Output . . . . .	181

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

<b>Chapter 12</b>	<b>Hot Plug Support</b>	<b>183</b>
12.1	Introduction	183
12.2	Hot Plug Features	183
12.3	Hot Plug Signals	183
12.4	Hot Plug Registers	183
12.5	Hot Plug Interrupts	183
12.6	Serial Hot Plug Controller	184
12.6.1	Hot Plug Operations by way of I <sup>2</sup> C I/O Expander	185
12.6.2	I <sup>2</sup> C I/O Expander Parts Selection and Pin Definition	185
12.6.3	Serial Hot Plug Port Enumeration and Assignment	189
<b>Chapter 13</b>	<b>Power Management</b>	<b>191</b>
13.1	Overview	191
13.2	Power Management Features	192
13.3	Power Management Capability	193
13.3.1	Device Power Management States	193
13.3.1.1	D0 Device Power Management State	193
13.3.1.2	D3hot Device Power Management State	193
13.3.2	Link Power Management States	193
13.3.3	PCI Express Power Management Support	195
13.4	Power Management Tracking	205
13.5	Power Management Event Handler	206
<b>Chapter 14</b>	<b>Transparent Port Registers</b>	<b>207</b>
14.1	Introduction	207
14.2	Type 1 Port Register Map	208
14.3	Port Register Configuration and Map	210
14.4	Register Access	213
14.4.1	PCI r3.0-Compatible Configuration Mechanism	214
14.4.2	PCI Express Enhanced Configuration Access Mechanism	215
14.4.3	Device-Specific Memory-Mapped Configuration Mechanism	215
14.5	Register Descriptions	217
14.6	PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)	218
14.7	PCI Power Management Capability Registers (Offsets 40h – 44h)	235
14.8	MSI Capability Registers (Offsets 48h – 64h)	238
14.9	PCI Express Capability Registers (Offsets 68h – A0h)	243
14.10	Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)	267
14.11	Device Serial Number Extended Capability Registers (Offsets 100h – 134h)	268
14.12	Power Budget Extended Capability Registers (Offsets 138h – 144h)	270
14.13	Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)	273
14.13.1	WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh)	280

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

14.14	Device-Specific Registers (Offsets 1C0h – 51Ch) . . . . .	283
14.14.1	Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh) . . . . .	284
14.14.2	Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch) . . . . .	320
14.14.3	Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch) . . . . .	416
14.14.4	Device-Specific Registers – Miscellaneous Control (Offset 28Ch) . . . . .	422
14.14.5	Device-Specific Registers – I <sup>2</sup> C Slave Interface (Offsets 290h – 2C4h) . . . . .	423
14.14.6	Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h) . . . . .	424
14.14.7	Device-Specific Registers – I/O CAM (Offsets 308h – 340h) . . . . .	431
14.14.8	Device-Specific Registers – SMBus Slave Interface (Offset 344h) . . . . .	440
14.14.9	Device-Specific Registers – Address-Mapping CAM (Offsets 348h – 444h) . . . . .	442
14.14.10	Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability (Offsets 448h – 51Ch) . . . . .	461
14.15	ACS Extended Capability Registers (Offsets 520h – 52Ch) . . . . .	482
14.16	Device-Specific Registers (Offsets 530h – F8Ch) . . . . .	487
14.16.1	Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h) . . . . .	489
14.16.2	Device-Specific Registers – Read Pacing (Offsets 544h – 554h) . . . . .	494
14.16.3	Device-Specific Registers – Port Configuration (Offsets 574h – 628h) . . . . .	502
14.16.4	Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch) . . . . .	504
14.16.5	Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch) . . . . .	523
14.16.6	Device-Specific Registers – IOCAM Base and Limit Upper 16 Bits (Offsets 680h – 6BCh) . . . . .	541
14.16.7	Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch) . . . . .	546
14.16.8	Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch) . . . . .	564
14.16.9	Device-Specific Registers – Ingress Credit Handler Port Pool (Offsets 940h – 94Ch) . . . . .	600
14.16.10	Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch) . . . . .	617
14.16.11	Device-Specific Registers – DMA Port Credit (Offsets 978h – 97Ch) . . . . .	619
14.16.12	Device-Specific Registers – ACS Extended Capability (Offsets 980h – 9FCh) . . . . .	622
14.16.13	Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch) . . . . .	704

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

14.16.14	Device-Specific Registers – Physical Layer (Offsets B80h – C30h) . . . . .	716
14.16.15	Device-Specific Registers – Port Configuration Header (Offsets E00h – E3Ch) . . . . .	743
14.16.16	Device-Specific Registers – Physical Layer (Offsets E40h – EFCh) . . . . .	792
14.16.17	Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch) . . . . .	796
14.16.18	Device-Specific Registers – Error Reporting (Offsets F40h – F4Ch) . . . . .	806
14.16.19	Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch) . . . . .	811
14.17	Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	820
<b>Chapter 15</b>	<b>Non-Transparent Bridging – NT Mode Only . . . . .</b>	<b>829</b>
15.1	Introduction . . . . .	829
15.1.1	Device Type Identification . . . . .	829
15.1.2	NT Port Features . . . . .	830
15.1.3	Intelligent Adapter Mode . . . . .	831
15.1.4	NT Port Reset . . . . .	832
15.1.4.1	Fundamental Reset (PEX_PERST#) . . . . .	832
15.1.4.2	Intelligent Adapter Mode NT Port Reset . . . . .	832
15.1.5	NT Port Memory-Mapped Base Address Registers . . . . .	833
15.1.6	Doorbell Registers . . . . .	835
15.1.7	Scratchpad Registers . . . . .	836
15.1.8	NT Base Address Registers . . . . .	836
15.1.8.1	NT BARx Setup Registers . . . . .	836
15.1.9	Address Translation . . . . .	838
15.1.9.1	Direct Address Translation . . . . .	839
15.2	NT PCI-to-PCI Bridge Mode . . . . .	841
15.3	Requester ID Translation . . . . .	842
15.3.1	Transaction Sequence . . . . .	843
15.3.2	Transaction Originating in Local Host Domain . . . . .	844
15.3.3	Transaction Originating in System Host Domain . . . . .	846
15.4	Traffic Class Translation . . . . .	848
15.5	NT Port Power Management Handling . . . . .	849
15.5.1	Active State Power Management . . . . .	849
15.5.2	PCI-PM and PME Turn Off Support . . . . .	849
15.5.3	Message Generation . . . . .	849
15.6	Expansion ROM . . . . .	850
15.7	NT Port Interrupts . . . . .	850
15.7.1	NT Port Virtual Interface Interrupts . . . . .	850
15.7.2	NT Port Link Interface Interrupts . . . . .	851
15.8	NT Port Error Handling . . . . .	852
15.8.1	NT Port Link Interface Error Handling . . . . .	853
15.8.2	NT PCI-to-PCI Bridge Mode Error Handling . . . . .	853
15.9	Cursor Mechanism . . . . .	854
15.10	Port Programmability . . . . .	854

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Chapter 16</b>	<b>NT Port Virtual Interface Registers – NT Mode Only.....</b>	<b>855</b>
16.1	Introduction .....	855
16.2	NT Port Virtual Interface Type 0 Register Map .....	856
16.3	Register Access .....	858
<b>16.3.1</b>	<i>PCI Express Base r2.0 Configuration Mechanism</i> .....	858
16.3.1.1	<i>PCI r3.0-Compatible Configuration Mechanism</i> .....	858
16.3.1.2	<i>PCI Express Enhanced Configuration Access Mechanism</i> .....	859
16.3.2	<i>Device-Specific Memory-Mapped Configuration Mechanism</i> .....	860
16.3.3	<i>Device-Specific Cursor Mechanism</i> .....	862
16.4	Register Descriptions .....	862
16.5	NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch) .....	863
16.6	NT Port Virtual Interface PCI Power Management Capability Registers (Offsets 40h – 44h) .....	873
16.7	NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h) .....	876
16.8	NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h) .....	877
16.9	NT Port Virtual Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h) .....	888
16.10	NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh) .....	889
16.11	NT Port Virtual Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h) .....	896
16.12	NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh) .....	897
16.12.1	NT Port Virtual Interface WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh) .....	899
16.13	NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h) .....	900
16.13.1	NT Port Virtual Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh) .....	903
16.13.2	NT Port Virtual Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch) .....	915
16.13.3	NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch) .....	917
16.13.4	NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h) .....	918
16.14	NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – EECh) .....	927
16.14.1	NT Port Virtual Interface NT Bridging-Specific Registers – Base Address and Base Address Setup (Offsets D68h – D90h) .....	928
16.14.2	NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DB0h) .....	936
16.15	NT Port Virtual Interface Device-Specific Registers (Offsets E40h – F8Ch) .....	938
16.15.1	NT Port Virtual Interface Device-Specific Registers – Physical Layer (Offsets E40h – EFCh) .....	939
16.15.2	NT Port Virtual Interface Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch) .....	939
16.15.3	NT Port Virtual Interface Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch) .....	940

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

16.16	NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	949
16.17	NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh) . . . . .	950
<b>Chapter 17</b>	<b>NT Port Link Interface Registers – NT Mode Only . . . . .</b>	<b>953</b>
17.1	Introduction . . . . .	953
17.2	NT Port Link Interface Type 0 Register Map . . . . .	954
17.3	Register Access . . . . .	956
<b>17.3.1 PCI Express Base r2.0 Configuration Mechanism . . . . .</b>	<b>956</b>	
17.3.1.1 PCI r3.0-Compatible Configuration Mechanism . . . . .	956	
17.3.1.2 PCI Express Enhanced Configuration Access Mechanism . . . . .	957	
17.3.2 Device-Specific Memory-Mapped Configuration Mechanism . . . . .	958	
17.3.3 Device-Specific Cursor Mechanism . . . . .	960	
17.4	Register Descriptions . . . . .	960
17.5	NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch) . . . . .	961
17.6	NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h) . . . . .	971
17.7	NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h) . . . . .	974
17.8	NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h) . . . . .	978
17.9	NT Port Link Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h) . . . . .	987
17.10	NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh) . . . . .	988
17.11	NT Port Link Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h) . . . . .	995
17.12	NT Port Link Interface Power Budget Extended Capability Registers (Offsets 138h – 144h) . . . . .	995
17.13	NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh) . . . . .	996
17.14	NT Port Link Interface Device-Specific Registers (Offsets 1C0h – C88h) . . . . .	997
17.14.1	NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh) . . . . .	998
17.14.2	NT Port Link Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch) . . . . .	1002
17.14.3	NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch) . . . . .	1003
17.14.4	NT Port Link Interface Device-Specific Registers – Ingress NT Control (Offset 960h) . . . . .	1004
17.14.5	NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h) . . . . .	1005
17.15	NT Port Link Interface NT Bridging-Specific Registers (Offsets C8Ch – EFCh) . . . . .	1007
17.15.1	NT Port Link Interface Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h) . . . . .	1008
17.16	NT Port Link Interface Device-Specific Registers – Source Queue Weight (Offsets F00h – F3Ch) . . . . .	1010
17.17	NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	1011

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Chapter 18</b>	<b>DMA Interface Registers – DMA Mode Only . . . . .</b>	<b>1013</b>
18.1	Introduction . . . . .	1013
18.2	DMA Interface Type 0 Register Map . . . . .	1014
18.3	Register Access . . . . .	1016
18.3.1	<i>PCI Express Base r2.0 Configuration Mechanism</i> . . . . .	1016
18.3.1.1	<i>PCI r3.0-Compatible Configuration Mechanism</i> . . . . .	1016
18.3.1.2	<i>PCI Express Enhanced Configuration Access Mechanism</i> . . . . .	1017
18.3.2	<i>Device-Specific Memory-Mapped Configuration Mechanism</i> . . . . .	1018
18.4	Register Descriptions . . . . .	1019
18.5	DMA Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch) . . . . .	1020
18.6	DMA Interface PCI Power Management Capability Registers (Offsets 40h – 44h) . . . . .	1029
18.7	DMA Interface MSI Capability Registers (Offsets 48h – 64h) . . . . .	1032
18.8	DMA Interface PCI Express Capability Registers (Offsets 68h – A0h) . . . . .	1036
18.9	DMA Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h) . . . . .	1046
18.10	DMA Interface Base Address DMA Setup Register (Offsets C8h – FCh) . . . . .	1047
18.11	DMA Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h) . . . . .	1048
18.12	Vendor-Specific Extended Capability Registers – DMA Channel (Offsets 1F4h – 5FCh) . . . . .	1050
18.13	DMA Device-Specific Registers – RAM Test and Soft Error (Offsets 600h – FB0h) . . . . .	1091
18.14	DMA Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	1093
<b>Chapter 19</b>	<b>Dual Clocking Support . . . . .</b>	<b>1101</b>
19.1	Introduction . . . . .	1101
19.2	Dual Clocking Operation . . . . .	1103
<b>Chapter 20</b>	<b>Test and Debug . . . . .</b>	<b>1105</b>
20.1	Introduction . . . . .	1105
20.2	Physical Layer Loopback Operation . . . . .	1106
20.2.1	Overview . . . . .	1106
20.2.2	Analog Loopback Master Mode . . . . .	1107
20.2.2.1	Initiating Far-End Analog Operations in PEX 8619 Master Devices . . . . .	1108
20.2.3	Digital Loopback Master Mode . . . . .	1110
20.2.4	Analog Loopback Slave Mode . . . . .	1111
20.2.5	Digital Loopback Slave Mode . . . . .	1114
20.3	User Test Pattern . . . . .	1115
20.4	Pseudo-Random Bit Sequence . . . . .	1117
20.4.1	Physical Layer Built-In Self-Test . . . . .	1119
20.5	Using the Even/Odd Port SerDes Quad $\times$ Diagnostic Data Registers . . . . .	1120
20.6	Pseudo-Random and Bit-Pattern Generation . . . . .	1121
20.7	PHY Testability Features . . . . .	1122

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

20.8 JTAG Interface .....	1124
20.8.1 <i>IEEE 1149.1 and IEEE 1149.6 Test Access Port</i> .....	1124
20.8.2 JTAG Instructions .....	1125
20.8.3 JTAG Boundary Scan .....	1126
20.8.4 JTAG Reset Input – JTAG_TRST# .....	1126
20.9 Lane Good Status LEDs .....	1127
<b>Chapter 21 Electrical Specifications .....</b>	<b>1129</b>
21.1 Introduction .....	1129
21.2 Power-Up/Power-Down Sequence .....	1129
21.3 Absolute Maximum Ratings .....	1129
21.4 Power Characteristics .....	1130
21.5 Power Consumption Estimates .....	1130
21.6 I/O Interface Signal Groupings .....	1131
21.7 Transmit Drive Characteristics .....	1141
21.7.1 Default Transmit Settings .....	1152
21.8 Receive Characteristics .....	1153
21.8.1 Receive Equalization .....	1153
21.8.2 Receiver Electrical Idle .....	1154
<b>Chapter 22 Thermal and Mechanical Specifications .....</b>	<b>1155</b>
22.1 Thermal Characteristics .....	1155
22.1.1 Sample Thermal Data .....	1155
22.2 General Package Specifications .....	1156
22.3 Mechanical Dimensions .....	1157
<b>Appendix A General Information .....</b>	<b>1159</b>
A.1 Product Ordering Information .....	1159
A.2 United States and International Representatives and Distributors .....	1160
A.3 Technical Support .....	1160

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

# Registers

## Transparent Port Registers

<b>PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch) . . . . .</b>	<b>218</b>
14-1. 00h PCI Configuration ID . . . . .	218
14-2. 04h PCI Command/Status . . . . .	219
14-3. 08h PCI Class Code and Revision ID . . . . .	222
14-4. 0Ch Miscellaneous Control. . . . .	223
14-5. 10h Base Address 0 . . . . .	224
14-6. 14h Base Address 1 . . . . .	225
14-7. 18h Bus Number. . . . .	225
14-8. 1Ch Secondary Status, I/O Limit, and I/O Base . . . . .	226
14-9. 20h Memory Base and Limit. . . . .	228
14-10. 24h Prefetchable Memory Base and Limit . . . . .	229
14-11. 28h Prefetchable Memory Upper Base Address . . . . .	230
14-12. 2Ch Prefetchable Memory Upper Limit Address . . . . .	230
14-13. 30h I/O Upper Base and Limit Address . . . . .	231
14-14. 34h Capability Pointer. . . . .	231
14-15. 38h Expansion ROM Base Address . . . . .	231
14-16. 3Ch Bridge Control and PCI Interrupt Signal . . . . .	232
<b>PCI Power Management Capability Registers (Offsets 40h – 44h) . . . . .</b>	<b>235</b>
14-17. 40h PCI Power Management Capability. . . . .	235
14-18. 44h PCI Power Management Status and Control. . . . .	236
<b>MSI Capability Registers (Offsets 48h – 64h) . . . . .</b>	<b>238</b>
14-19. 48h MSI Capability. . . . .	239
14-20. 4Ch MSI Address . . . . .	240
14-21. 50h MSI Upper Address . . . . .	240
14-22. 54h MSI Data . . . . .	240
14-23. 58h MSI Mask. . . . .	241
14-24. 5Ch MSI Status . . . . .	242
<b>PCI Express Capability Registers (Offsets 68h – A0h) . . . . .</b>	<b>243</b>
14-25. 68h PCI Express Capability List and Capability . . . . .	244
14-26. 6Ch Device Capability . . . . .	245
14-27. 70h Device Status and Control. . . . .	247
14-28. 74h Link Capability . . . . .	249
14-29. 78h Link Status and Control . . . . .	252
14-30. 7Ch Slot Capability . . . . .	255
14-31. 80h Slot Status and Control . . . . .	258
14-32. 8Ch Device Capability 2 . . . . .	264
14-33. 90h Device Status and Control 2 . . . . .	264
14-34. 98h Link Status and Control 2 . . . . .	265
<b>Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh) . . . . .</b>	<b>267</b>
14-35. A4h Subsystem Capability . . . . .	267
14-36. A8h Subsystem ID and Subsystem Vendor ID . . . . .	267

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

<b>Device Serial Number Extended Capability Registers</b>	
<b>(Offsets 100h – 134h) . . . . .</b>	<b>268</b>
14-37. 100h Device Serial Number Extended Capability Header . . . . .	268
14-38. 104h Serial Number (Lower DW) . . . . .	269
14-39. 108h Serial Number (Upper DW) . . . . .	269
<b>Power Budget Extended Capability Registers</b>	
<b>(Offsets 138h – 144h) . . . . .</b>	<b>270</b>
14-40. 138h Power Budget Extended Capability Header . . . . .	270
14-41. 13Ch Data Select . . . . .	271
14-42. 140h Power Budget Data . . . . .	271
14-43. 144h Power Budget Capability . . . . .	272
<b>Virtual Channel Extended Capability Registers</b>	
<b>(Offsets 148h – 1BCh) . . . . .</b>	<b>273</b>
14-44. 148h Virtual Channel Extended Capability Header . . . . .	273
14-45. 14Ch Port VC Capability 1 . . . . .	274
14-46. 150h Port VC Capability 2 . . . . .	275
14-47. 154h Port VC Status and Control . . . . .	275
14-48. 158h VC0 Resource Capability . . . . .	276
14-49. 15Ch VC0 Resource Control . . . . .	277
14-50. 160h VC0 Resource Status . . . . .	277
14-51. 164h VC1 Resource Capability . . . . .	278
14-52. 168h VC1 Resource Control . . . . .	278
14-53. 16Ch VC1 Resource Status . . . . .	279
<b>WRR Port Arbitration Table Registers</b>	
<b>(Offsets 1A8h – 1BCh) . . . . .</b>	<b>280</b>
14-54. 1A8h WRR Port Arbitration Table Phases 0 to 7 . . . . .	281
14-55. 1ACh WRR Port Arbitration Table Phases 8 to 15 . . . . .	281
14-56. 1B0h WRR Port Arbitration Table Phases 16 to 23 . . . . .	282
14-57. 1B4h WRR Port Arbitration Table Phases 24 to 31 . . . . .	282
<b>Device-Specific Registers</b>	
<b>(Offsets 1C0h – 51Ch) . . . . .</b>	<b>283</b>
<b>Device-Specific Registers – Error Checking and Debug</b>	
<b>(Offsets 1C0h – 1FCh) . . . . .</b>	<b>284</b>
14-58. 1C0h Device-Specific Error Status for Egress ECC Error . . . . .	285
14-59. 1C4h Device-Specific Error Mask for Egress ECC Error . . . . .	289
14-60. 1C8h ECC Error Check Disable . . . . .	293
14-61. 1CCh Error Handler 32-Bit Error Status . . . . .	295
14-62. 1D0h Error Handler 32-Bit Error Mask . . . . .	297
14-63. 1D8h Clock Enable . . . . .	300
14-64. 1DCh Debug Control . . . . .	303
14-65. 1E0h Power Management Hot Plug User Configuration . . . . .	312
14-66. 1E4h Egress Control and Status . . . . .	314
14-67. 1E8h Bad TLP Counter . . . . .	316
14-68. 1ECh Bad DLLP Counter . . . . .	316
14-69. 1F4h Software Lane Status . . . . .	317
14-70. 1F8h ACK Transmission Latency Limit . . . . .	319

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch).....</b>	<b>320</b>
14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status .....	322
14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status .....	328
14-73. 210h Physical Layer User Test Pattern, Bytes 0 through 3.....	333
14-74. 214h Physical Layer User Test Pattern, Bytes 4 through 7.....	333
14-75. 218h Physical Layer User Test Pattern, Bytes 8 through 11.....	334
14-76. 21Ch Physical Layer User Test Pattern, Bytes 12 through 15 .....	334
14-77. 220h Even Ports – Physical Layer Command/Status .....	335
14-78. 224h Odd Ports – Physical Layer Command/Status.....	343
14-79. 228h Even Ports – Physical Layer Test.....	351
14-80. 22Ch Odd Ports – Physical Layer Test .....	357
14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate .....	363
14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate .....	370
14-83. 238h Common Physical Layer Command/Status .....	377
14-84. 23Ch Common Physical Layer Safety and Test .....	378
14-85. 240h Even Ports – SerDes Quad 0 Diagnostic Data .....	380
14-86. 244h Odd Ports – SerDes Quad 2 Diagnostic Data .....	382
14-87. 248h Even Ports – SerDes Quad 1 Diagnostic Data .....	384
14-88. 24Ch Odd Ports – SerDes Quad 3 Diagnostic Data.....	386
14-89. 250h Even Ports – Physical Layer Safety and Test .....	388
14-90. 254h Odd Ports – Physical Layer Safety and Test .....	392
14-91. 258h Even Ports – User Test Pattern Delay .....	396
14-92. 25Ch Odd Ports – User Test Pattern Delay .....	406
<b>Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch).....</b>	<b>416</b>
14-93. 260h Serial EEPROM Status and Control .....	416
14-94. 264h Serial EEPROM Buffer .....	419
14-95. 268h Serial EEPROM Clock Frequency .....	420
14-96. 26Ch Serial EEPROM 3 <sup>rd</sup> Address Byte .....	421
<b>Device-Specific Registers – Miscellaneous Control (Offset 28Ch).....</b>	<b>422</b>
14-97. 28Ch Miscellaneous Control.....	422
<b>Device-Specific Registers – I<sup>2</sup>C Slave Interface (Offsets 290h – 2C4h).....</b>	<b>423</b>
14-98. 294h I <sup>2</sup> C Configuration .....	423
<b>Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h).....</b>	<b>424</b>
14-99. 2C8h BusNoCAM0 .....	425
14-100. 2CCh BusNoCAM1 .....	425
14-101. 2D0h BusNoCAM2 .....	425
14-102. 2D4h BusNoCAM3 .....	426
14-103. 2D8h BusNoCAM4 .....	426
14-104. 2DCh BusNoCAM5 .....	426
14-105. 2E0h BusNoCAM6 .....	427
14-106. 2E4h BusNoCAM7 .....	427
14-107. 2E8h BusNoCAM8 .....	427
14-108. 2ECh BusNoCAM9 .....	428
14-109. 2F0h BusNoCAM10 .....	428
14-110. 2F4h BusNoCAM11 .....	428
14-111. 2F8h BusNoCAM12 .....	429
14-112. 2FCh BusNoCAM13 .....	429

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

14-113. 300h BusNoCAM14.....	429
14-114. 304h BusNoCAM15.....	430

**Device-Specific Registers – I/O CAM  
(Offsets 308h – 340h) .....** **431**

14-115. 308h IOCAM0 .....	432
14-116. 30Ah IOCAM1 .....	432
14-117. 30Ch IOCAM2.....	433
14-118. 30Eh IOCAM3 .....	433
14-119. 310h IOCAM4 .....	434
14-120. 312h IOCAM5 .....	434
14-121. 314h IOCAM6 .....	435
14-122. 316h IOCAM7 .....	435
14-123. 318h IOCAM8 .....	436
14-124. 31Ah IOCAM9.....	436
14-125. 31Ch IOCAM10.....	437
14-126. 31Eh IOCAM11.....	437
14-127. 320h IOCAM12 .....	438
14-128. 322h IOCAM13 .....	438
14-129. 324h IOCAM14 .....	439
14-130. 326h IOCAM15 .....	439

**Device-Specific Registers – SMBus Slave Interface  
(Offset 344h) .....** **440**

14-131. 344h SMBus Configuration .....	440
--	-----

**Device-Specific Registers – Address-Mapping CAM  
(Offsets 348h – 444h) .....** **442**

14-132. 348h AMCAM0 Memory Base and Limit .....	445
14-133. 34Ch AMCAM0 Prefetchable Memory Base and Limit .....	445
14-134. 350h AMCAM0 Prefetchable Memory Base Upper 32 Bits.....	445
14-135. 354h AMCAM0 Prefetchable Memory Limit Upper 32 Bits .....	445
14-136. 358h AMCAM1 Memory Base and Limit .....	446
14-137. 35Ch AMCAM1 Prefetchable Memory Base and Limit .....	446
14-138. 360h AMCAM1 Prefetchable Memory Base Upper 32 Bits.....	446
14-139. 364h AMCAM1 Prefetchable Memory Limit Upper 32 Bits .....	446
14-140. 368h AMCAM2 Memory Base and Limit .....	447
14-141. 36Ch AMCAM2 Prefetchable Memory Base and Limit .....	447
14-142. 370h AMCAM2 Prefetchable Memory Base Upper 32 Bits.....	447
14-143. 374h AMCAM2 Prefetchable Memory Limit Upper 32 Bits .....	447
14-144. 378h AMCAM3 Memory Base and Limit .....	448
14-145. 37Ch AMCAM3 Prefetchable Memory Base and Limit .....	448
14-146. 380h AMCAM3 Prefetchable Memory Base Upper 32 Bits.....	448
14-147. 384h AMCAM3 Prefetchable Memory Limit Upper 32 Bits .....	448
14-148. 388h AMCAM4 Memory Base and Limit .....	449
14-149. 38Ch AMCAM4 Prefetchable Memory Base and Limit .....	449
14-150. 390h AMCAM4 Prefetchable Memory Base Upper 32 Bits.....	449
14-151. 394h AMCAM4 Prefetchable Memory Limit Upper 32 Bits .....	449
14-152. 398h AMCAM5 Memory Base and Limit .....	450
14-153. 39Ch AMCAM5 Prefetchable Memory Base and Limit .....	450
14-154. 3A0h AMCAM5 Prefetchable Memory Base Upper 32 Bits.....	450
14-155. 3A4h AMCAM5 Prefetchable Memory Limit Upper 32 Bits .....	450
14-156. 3A8h AMCAM6 Memory Base and Limit .....	451
14-157. 3ACh AMCAM6 Prefetchable Memory Base and Limit .....	451
14-158. 3B0h AMCAM6 Prefetchable Memory Base Upper 32 Bits.....	451
14-159. 3B4h AMCAM6 Prefetchable Memory Limit Upper 32 Bits .....	451
14-160. 3B8h AMCAM7 Memory Base and Limit .....	452
14-161. 3BCh AMCAM7 Prefetchable Memory Base and Limit .....	452

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

14-162. 3C0h AMCAM7 Prefetchable Memory Base Upper 32 Bits . . . . .	452
14-163. 3C4h AMCAM7 Prefetchable Memory Limit Upper 32 Bits . . . . .	452
14-164. 3C8h AMCAM8 Memory Base and Limit . . . . .	453
14-165. 3CCh AMCAM8 Prefetchable Memory Base and Limit . . . . .	453
14-166. 3D0h AMCAM8 Prefetchable Memory Base Upper 32 Bits . . . . .	453
14-167. 3D4h AMCAM8 Prefetchable Memory Limit Upper 32 Bits . . . . .	453
14-168. 3D8h AMCAM9 Memory Base and Limit . . . . .	454
14-169. 3DCh AMCAM9 Prefetchable Memory Base and Limit . . . . .	454
14-170. 3E0h AMCAM9 Prefetchable Memory Base Upper 32 Bits . . . . .	454
14-171. 3E4h AMCAM9 Prefetchable Memory Limit Upper 32 Bits . . . . .	454
14-172. 3E8h AMCAM10 Memory Base and Limit . . . . .	455
14-173. 3ECh AMCAM10 Prefetchable Memory Base and Limit . . . . .	455
14-174. 3F0h AMCAM10 Prefetchable Memory Base Upper 32 Bits . . . . .	455
14-175. 3F4h AMCAM10 Prefetchable Memory Limit Upper 32 Bits. . . . .	455
14-176. 3F8h AMCAM11 Memory Base and Limit . . . . .	456
14-177. 3FCh AMCAM11 Prefetchable Memory Base and Limit. . . . .	456
14-178. 400h AMCAM11 Prefetchable Memory Base Upper 32 Bits . . . . .	456
14-179. 404h AMCAM11 Prefetchable Memory Limit Upper 32 Bits. . . . .	456
14-180. 408h AMCAM12 Memory Base and Limit . . . . .	457
14-181. 40Ch AMCAM12 Prefetchable Memory Base and Limit. . . . .	457
14-182. 410h AMCAM12 Prefetchable Memory Base Upper 32 Bits . . . . .	457
14-183. 414h AMCAM12 Prefetchable Memory Limit Upper 32 Bits. . . . .	457
14-184. 418h AMCAM13 Memory Base and Limit . . . . .	458
14-185. 41Ch AMCAM13 Prefetchable Memory Base and Limit. . . . .	458
14-186. 420h AMCAM13 Prefetchable Memory Base Upper 32 Bits . . . . .	458
14-187. 424h AMCAM13 Prefetchable Memory Limit Upper 32 Bits. . . . .	458
14-188. 428h AMCAM14 Memory Base and Limit . . . . .	459
14-189. 42Ch AMCAM14 Prefetchable Memory Base and Limit. . . . .	459
14-190. 430h AMCAM14 Prefetchable Memory Base Upper 32 Bits . . . . .	459
14-191. 434h AMCAM14 Prefetchable Memory Limit Upper 32 Bits. . . . .	459
14-192. 438h AMCAM15 Memory Base and Limit . . . . .	460
14-193. 43Ch AMCAM15 Prefetchable Memory Base and Limit. . . . .	460
14-194. 440h AMCAM15 Prefetchable Memory Base Upper 32 Bits . . . . .	460
14-195. 444h AMCAM15 Prefetchable Memory Limit Upper 32 Bits. . . . .	460

**Device-Specific Registers – Vendor-Specific Dual Cast**

<b>Extended Capability (Offsets 448h – 51Ch) . . . . .</b>	<b>461</b>
14-196. 448h Vendor-Specific Extended Capability Header . . . . .	463
14-197. 44Ch Vendor-Specific Header 1 . . . . .	463
14-198. 450h Dual Cast Low BAR0. . . . .	464
14-199. 454h Dual Cast High BAR0 . . . . .	464
14-200. 458h Dual Cast Low BAR0 Translation. . . . .	464
14-201. 45Ch Dual Cast High BAR0 Translation . . . . .	464
14-202. 460h Dual Cast Low BAR0 Setup. . . . .	465
14-203. 464h Dual Cast High BAR0 Setup . . . . .	465
14-204. 468h Dual Cast Low BAR1. . . . .	466
14-205. 46Ch Dual Cast High BAR1 . . . . .	466
14-206. 470h Dual Cast Low BAR1 Translation. . . . .	466
14-207. 474h Dual Cast High BAR1 Translation . . . . .	466
14-208. 478h Dual Cast Low BAR1 Setup. . . . .	467
14-209. 47Ch Dual Cast High BAR1 Setup. . . . .	467
14-210. 480h Dual Cast Low BAR2. . . . .	468
14-211. 484h Dual Cast High BAR2 . . . . .	468
14-212. 488h Dual Cast Low BAR2 Translation. . . . .	468
14-213. 48Ch Dual Cast High BAR2 Translation . . . . .	468
14-214. 490h Dual Cast Low BAR2 Setup. . . . .	469
14-215. 494h Dual Cast High BAR2 Setup . . . . .	469
14-216. 498h Dual Cast Low BAR3. . . . .	470
14-217. 49Ch Dual Cast High BAR3 . . . . .	470

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

14-218. 4A0h Dual Cast Low BAR3 Translation . . . . .	470
14-219. 4A4h Dual Cast High BAR3 Translation . . . . .	470
14-220. 4A8h Dual Cast Low BAR3 Setup . . . . .	471
14-221. 4ACh Dual Cast High BAR3 Setup . . . . .	471
14-222. 4B0h Dual Cast Low BAR4 . . . . .	472
14-223. 4B4h Dual Cast High BAR4 . . . . .	472
14-224. 4B8h Dual Cast Low BAR4 Translation . . . . .	472
14-225. 4BCh Dual Cast High BAR4 Translation . . . . .	472
14-226. 4C0h Dual Cast Low BAR4 Setup . . . . .	473
14-227. 4C4h Dual Cast High BAR4 Setup . . . . .	473
14-228. 4C8h Dual Cast Low BAR5 . . . . .	474
14-229. 4CCh Dual Cast High BAR5 . . . . .	474
14-230. 4D0h Dual Cast Low BAR5 Translation . . . . .	474
14-231. 4D4h Dual Cast High BAR5 Translation . . . . .	474
14-232. 4D8h Dual Cast Low BAR5 Setup . . . . .	475
14-233. 4DCh Dual Cast High BAR5 Setup . . . . .	475
14-234. 4E0h Dual Cast Low BAR6 . . . . .	476
14-235. 4E4h Dual Cast High BAR6 . . . . .	476
14-236. 4E8h Dual Cast Low BAR6 Translation . . . . .	476
14-237. 4ECH Dual Cast High BAR6 Translation . . . . .	476
14-238. 4F0h Dual Cast Low BAR6 Setup . . . . .	477
14-239. 4F4h Dual Cast High BAR6 Setup . . . . .	477
14-240. 4F8h Dual Cast Low BAR7 . . . . .	478
14-241. 4FCh Dual Cast High BAR7 . . . . .	478
14-242. 500h Dual Cast Low BAR7 Translation . . . . .	478
14-243. 504h Dual Cast High BAR7 Translation . . . . .	478
14-244. 508h Dual Cast Low BAR7 Setup . . . . .	479
14-245. 50Ch Dual Cast High BAR7 Setup . . . . .	479
14-246. 510h Dual Cast Source Destination Port . . . . .	480
<b>ACS Extended Capability Registers (Offsets 520h – 52Ch) . . . . .</b>	<b>482</b>
14-247. 520h ACS Extended Capability Header . . . . .	482
14-248. 524h ACS Control and Capability . . . . .	483
14-249. 528h Egress Control Vector . . . . .	485
<b>Device-Specific Registers (Offsets 530h – F8Ch) . . . . .</b>	<b>487</b>
<b>Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h) . . . . .</b>	<b>489</b>
14-250. 530h VC WRR 0 . . . . .	490
14-251. 534h VC WRR 1 . . . . .	490
14-252. 538h VC WRR 2 . . . . .	491
14-253. 53Ch VC WRR 3 . . . . .	491
14-254. 540h VC Fixed Arbitration Select . . . . .	492
<b>Device-Specific Registers – Read Pacing (Offsets 544h – 554h) . . . . .</b>	<b>494</b>
14-255. 544h Read Pacing Control . . . . .	495
14-256. 548h Read Spreading Control . . . . .	497
14-257. 54Ch Read Pacing Threshold 0 . . . . .	499
14-258. 550h Read Pacing Threshold 1 . . . . .	499
14-259. 554h Read Counter Clear . . . . .	500
<b>Device-Specific Registers – Port Configuration (Offsets 574h – 628h) . . . . .</b>	<b>502</b>
14-260. 574h Port Configuration . . . . .	503

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch) .....</b>	<b>504</b>
14-261. 62Ch GPIO 0_15 Direction Control .....	505
14-262. 630h GPIO 16_31 Direction Control.....	509
14-263. 638h GPIO 0_31 Input De-Bounce.....	513
14-264. 63Ch GPIO 0_15 Input Data .....	514
14-265. 640h GPIO 16_31 Input Data.....	516
14-266. 644h GPIO 0_15 Output Data .....	518
14-267. 648h GPIO 16_31 Output Data .....	520
14-268. 64Ch GPIO 0_31 Interrupt Polarity.....	522
14-269. 650h GPIO 0_31 Interrupt Status.....	522
14-270. 654h GPIO 0_31 Interrupt Mask.....	522
<b>Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch) .....</b>	<b>523</b>
14-271. 660h Ingress Control .....	524
14-272. 664h Ingress Control Shadow .....	525
14-273. 668h Port Enable Status.....	527
14-274. 66Ch Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7 .....	532
14-275. 670h Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15 .....	534
14-276. 674h Port Cut-Thru Enable Status .....	536
14-277. 678h Ingress PLL RAM ECC 1-Bit Counter .....	539
14-278. 67Ch Ingress Control 2.....	540
<b>Device-Specific Registers – IOCAM Base and Limit</b>	
<b>Upper 16 Bits (Offsets 680h – 6BCh) .....</b>	<b>541</b>
14-279. 680h IOCAM UP 0 .....	542
14-280. 684h IOCAM UP 1 .....	542
14-281. 688h IOCAM UP 2 .....	542
14-282. 68Ch IOCAM UP 3 .....	542
14-283. 690h IOCAM UP 4 .....	543
14-284. 694h IOCAM UP 5 .....	543
14-285. 698h IOCAM UP 6 .....	543
14-286. 69Ch IOCAM UP 7 .....	543
14-287. 6A0h IOCAM UP 8 .....	544
14-288. 6A4h IOCAM UP 9 .....	544
14-289. 6A8h IOCAM UP 10 .....	544
14-290. 6ACh IOCAM UP 11 .....	544
14-291. 6B0h IOCAM UP 12 .....	545
14-292. 6B4h IOCAM UP 13 .....	545
14-293. 6B8h IOCAM UP 14 .....	545
14-294. 6BCh IOCAM UP 15 .....	545
<b>Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch) .....</b>	<b>546</b>
14-295. 6C0h BAR0 Shadow for Port 0 .....	548
14-296. 6C4h BAR1 Shadow for Port 0 .....	548
14-297. 6C8h BAR0 Shadow for Port 1 .....	549
14-298. 6CCh BAR1 Shadow for Port 1 .....	549
14-299. 6D0h BAR0 Shadow for Port 2 .....	550
14-300. 6D4h BAR1 Shadow for Port 2 .....	550
14-301. 6D8h BAR0 Shadow for Port 3 .....	551
14-302. 6DCh BAR1 Shadow for Port 3 .....	551
14-303. 6E0h BAR0 Shadow for Port 4 .....	552
14-304. 6E4h BAR1 Shadow for Port 4 .....	552
14-305. 6E8h BAR0 Shadow for Port 5 .....	553
14-306. 6ECh BAR1 Shadow for Port 5 .....	553

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

14-307. 6F0h BAR0 Shadow for Port 6 .....	554
14-308. 6F4h BAR1 Shadow for Port 6 .....	554
14-309. 6F8h BAR0 Shadow for Port 7 .....	555
14-310. 6FC <sub>h</sub> BAR1 Shadow for Port 7 .....	555
14-311. 700h BAR0 Shadow for Port 8 .....	556
14-312. 704h BAR1 Shadow for Port 8 .....	556
14-313. 708h BAR0 Shadow for Port 9 .....	557
14-314. 70Ch BAR1 Shadow for Port 9 .....	557
14-315. 710h BAR0 Shadow for Port 10 .....	558
14-316. 714h BAR1 Shadow for Port 10 .....	558
14-317. 718h BAR0 Shadow for Port 11 .....	559
14-318. 71Ch BAR1 Shadow for Port 11 .....	559
14-319. 720h BAR0 Shadow for Port 12 .....	560
14-320. 724h BAR1 Shadow for Port 12 .....	560
14-321. 728h BAR0 Shadow for Port 13 .....	561
14-322. 72Ch BAR1 Shadow for Port 13 .....	561
14-323. 730h BAR0 Shadow for Port 14 .....	562
14-324. 734h BAR1 Shadow for Port 14 .....	562
14-325. 738h BAR0 Shadow for Port 15 .....	563
14-326. 73Ch BAR1 Shadow for Port 15 .....	563
<b>Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch).....</b>	<b>564</b>
14-327. 740h Port 0 VC0 Resource Control .....	566
14-328. 744h Port 0 VC1 Resource Control .....	567
14-329. 748h Port 1 VC0 Resource Control .....	568
14-330. 74Ch Port 1 VC1 Resource Control .....	569
14-331. 750h Port 2 VC0 Resource Control .....	570
14-332. 754h Port 2 VC1 Resource Control .....	571
14-333. 758h Port 3 VC0 Resource Control .....	572
14-334. 75Ch Port 3 VC1 Resource Control .....	573
14-335. 760h Port 4 VC0 Resource Control .....	574
14-336. 764h Port 4 VC1 Resource Control .....	575
14-337. 768h Port 5 VC0 Resource Control .....	576
14-338. 76Ch Port 5 VC1 Resource Control .....	577
14-339. 770h Port 6 VC0 Resource Control .....	578
14-340. 774h Port 6 VC1 Resource Control .....	579
14-341. 778h Port 7 VC0 Resource Control .....	580
14-342. 77Ch Port 7 VC1 Resource Control .....	581
14-343. 780h Port 8 VC0 Resource Control .....	582
14-344. 784h Port 8 VC1 Resource Control .....	583
14-345. 788h Port 9 VC0 Resource Control .....	584
14-346. 78Ch Port 9 VC1 Resource Control .....	585
14-347. 790h Port 10 VC0 Resource Control .....	586
14-348. 794h Port 10 VC1 Resource Control .....	587
14-349. 798h Port 11 VC0 Resource Control .....	588
14-350. 79Ch Port 11 VC1 Resource Control .....	589
14-351. 7A0h Port 12 VC0 Resource Control .....	590
14-352. 7A4h Port 12 VC1 Resource Control .....	591
14-353. 7A8h Port 13 VC0 Resource Control .....	592
14-354. 7ACh Port 13 VC1 Resource Control .....	593
14-355. 7B0h Port 14 VC0 Resource Control .....	594
14-356. 7B4h Port 14 VC1 Resource Control .....	595
14-357. 7B8h Port 15 VC0 Resource Control .....	596
14-358. 7BC <sub>h</sub> Port 15 VC1 Resource Control .....	597

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

14-359. 7C0h Base Address 0 DMA .....	598
14-360. 7C4h Base Address 1 DMA .....	598
14-361. 7C8h DMA Minimum and Maximum Read Latency .....	599
14-362. 7CCh DMA Minimum and Maximum Write Latency .....	599
<b>Device-Specific Registers – Ingress Credit Handler Port Pool</b>	
<b>(Offsets 940h – 94Ch) .....</b>	<b>600</b>
14-363. 940h INCH Port Pool for Ports 0, 1, 2, 3 .....	601
14-364. 944h INCH Port Pool for Ports 4, 5, 6, 7 .....	605
14-365. 948h INCH Port Pool for Ports 8, 9, 10, 11 .....	609
14-366. 94Ch INCH Port Pool for Ports 12, 13, 14, 15 .....	613
<b>Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)</b> .....	<b>617</b>
14-367. 950h Vendor-Specific Extended Capability 2 .....	617
14-368. 954h Vendor-Specific Header 2 .....	617
14-369. 958h PLX Hardwired Configuration ID .....	618
14-370. 95Ch PLX Hardwired Revision ID .....	618
<b>Device-Specific Registers – DMA Port Credit</b>	
<b>(Offsets 978h – 97Ch) .....</b>	<b>619</b>
14-371. 978h Even Ports – DMA Port Credit .....	620
14-372. 97Ch Odd Ports – DMA Port Credit .....	621
<b>Device-Specific Registers – ACS Extended Capability</b>	
<b>(Offsets 980h – 9FCh) .....</b>	<b>622</b>
14-373. 980h ACS Port 0 Control and Capability .....	624
14-374. 984h ACS Port 0 Egress Control Vector .....	627
14-375. 988h ACS Port 1 Control and Capability .....	629
14-376. 98Ch ACS Port 1 Egress Control Vector .....	632
14-377. 990h ACS Port 2 Control and Capability .....	634
14-378. 994h ACS Port 2 Egress Control Vector .....	637
14-379. 998h ACS Port 3 Control and Capability .....	639
14-380. 99Ch ACS Port 3 Egress Control Vector .....	642
14-381. 9A0h ACS Port 4 Control and Capability .....	644
14-382. 9A4h ACS Port 4 Egress Control Vector .....	647
14-383. 9A8h ACS Port 5 Control and Capability .....	649
14-384. 9ACh ACS Port 5 Egress Control Vector .....	652
14-385. 9B0h ACS Port 6 Control and Capability .....	654
14-386. 9B4h ACS Port 6 Egress Control Vector .....	657
14-387. 9B8h ACS Port 7 Control and Capability .....	659
14-388. 9BCh ACS Port 7 Egress Control Vector .....	662
14-389. 9C0h ACS Port 8 Control and Capability .....	664
14-390. 9C4h ACS Port 8 Egress Control Vector .....	667
14-391. 9C8h ACS Port 9 Control and Capability .....	669
14-392. 9CCh ACS Port 9 Egress Control Vector .....	672
14-393. 9D0h ACS Port 10 Control and Capability .....	674
14-394. 9D4h ACS Port 10 Egress Control Vector .....	677
14-395. 9D8h ACS Port 11 Control and Capability .....	679
14-396. 9DCh ACS Port 11 Egress Control Vector .....	682
14-397. 9E0h ACS Port 12 Control and Capability .....	684
14-398. 9E4h ACS Port 12 Egress Control Vector .....	687
14-399. 9E8h ACS Port 13 Control and Capability .....	689
14-400. 9ECh ACS Port 13 Egress Control Vector .....	692
14-401. 9F0h ACS Port 14 Control and Capability .....	694
14-402. 9F4h ACS Port 14 Egress Control Vector .....	697
14-403. 9F8h ACS Port 15 Control and Capability .....	699
14-404. 9FCh ACS Port 15 Egress Control Vector .....	702

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

<b>Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch) . . . . .</b>	<b>704</b>
14-405. A00h, A18h, A30h, A48h, A60h, A78h, A90h, AA8h, AC0h, AD8h, AF0h, B08h, B20h, B38h, B50h, B68h INCH Threshold Port x VC0 Posted . . . . .	707
14-406. A04h, A1Ch, A34h, A4Ch, A64h, A7Ch, A94h, AACh, AC4h, ADCh, AF4h, B0Ch, B24h, B3Ch, B54h, B6Ch INCH Threshold Port x VC0 Non-Posted . . . . .	709
14-407. A08h, A20h, A38h, A50h, A68h, A80h, A98h, AB0h, AC8h, AE0h, AF8h, B10h, B28h, B40h, B58h, B70h INCH Threshold Port x VC0 Completion . . . . .	711
14-408. A0Ch, A24h, A3Ch, A54h, A6Ch, A84h, A9Ch, AB4h, ACCh, AE4h, AFCh, B14h, B2Ch, B44h, B5Ch, B74h INCH Threshold Port x VC1 Posted . . . . .	712
14-409. A10h, A28h, A40h, A58h, A70h, A88h, AA0h, AB8h, AD0h, AE8h, B00h, B18h, B30h, B48h, B60h, B78h INCH Threshold Port x VC1 Non-Posted . . . . .	714
14-410. A14h, A2Ch, A44h, A5Ch, A74h, A8Ch, AA4h, ABCh, AD4h, AECh, B04h, B1Ch, B34h, B4Ch, B64h, B7Ch INCH Threshold Port x VC1 Completion . . . . .	715
<b>Device-Specific Registers – Physical Layer (Offsets B80h – C30h) . . . . .</b>	<b>716</b>
14-411. B80h Even Ports – Loopback Master Status/Control . . . . .	717
14-412. B84h Odd Ports – Loopback Master Status/Control . . . . .	722
14-413. B88h Even Ports 0, 2, 4, 6 – Receiver Error Counters . . . . .	726
14-414. B8Ch Odd Ports 1, 3, 5, 7 – Receiver Error Counters . . . . .	726
14-415. B90h Even Ports 8, 10, 12, 14 – Receiver Error Counters . . . . .	727
14-416. B94h Odd Ports 9, 11, 13, 15 – Receiver Error Counters . . . . .	727
14-417. B98h SerDes Drive Level 0 . . . . .	728
14-418. B9Ch SerDes Drive Level 1 . . . . .	729
14-419. BA0h SerDes Drive Level 2 . . . . .	730
14-420. BA4h SerDes Drive Level 3 . . . . .	731
14-421. BA8h Post-Cursor Emphasis Level 0 . . . . .	732
14-422. BACh Post-Cursor Emphasis Level 1 . . . . .	733
14-423. BB0h Post-Cursor Emphasis Level 2 . . . . .	734
14-424. BB4h Post-Cursor Emphasis Level 3 . . . . .	735
14-425. BB8h Receiver Equalization Level 0 . . . . .	736
14-426. BBCh Receiver Equalization Level 1 . . . . .	736
14-427. BC0h Signal Detect Level . . . . .	737
14-428. C28h SerDes Control . . . . .	738
<b>Device-Specific Registers – Port Configuration Header (Offsets E00h – E3Ch) . . . . .</b>	<b>743</b>
14-429. E00h PCI Command Port 0 . . . . .	744
14-430. E02h PCI Command Port 1 . . . . .	745
14-431. E04h PCI Command Port 2 . . . . .	746
14-432. E06h PCI Command Port 3 . . . . .	747
14-433. E08h PCI Command Port 4 . . . . .	748
14-434. E0Ah PCI Command Port 5 . . . . .	749
14-435. E0Ch PCI Command Port 6 . . . . .	750
14-436. E0Eh PCI Command Port 7 . . . . .	751
14-437. E10h PCI Command Port 8 . . . . .	752
14-438. E12h PCI Command Port 9 . . . . .	753
14-439. E14h PCI Command Port 10 . . . . .	754
14-440. E16h PCI Command Port 11 . . . . .	755
14-441. E18h PCI Command Port 12 . . . . .	756
14-442. E1Ah PCI Command Port 13 . . . . .	757
14-443. E1Ch PCI Command Port 14 . . . . .	758
14-444. E1Eh PCI Command Port 15 . . . . .	759

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

14-445. E20h Bridge Control Port 0 .....	760
14-446. E22h Bridge Control Port 1 .....	762
14-447. E24h Bridge Control Port 2 .....	764
14-448. E26h Bridge Control Port 3 .....	766
14-449. E28h Bridge Control Port 4 .....	768
14-450. E2Ah Bridge Control Port 5 .....	770
14-451. E2Ch Bridge Control Port 6 .....	772
14-452. E2Eh Bridge Control Port 7 .....	774
14-453. E30h Bridge Control Port 8 .....	776
14-454. E32h Bridge Control Port 9 .....	778
14-455. E34h Bridge Control Port 10 .....	780
14-456. E36h Bridge Control Port 11 .....	782
14-457. E38h Bridge Control Port 12 .....	784
14-458. E3Ah Bridge Control Port 13 .....	786
14-459. E3Ch Bridge Control Port 14 .....	788
14-460. E3Eh Bridge Control Port 15 .....	790
<b>Device-Specific Registers – Physical Layer</b>	
<b>(Offsets E40h – EFCh) .....</b>	<b>792</b>
14-461. E40h CSR-Based SerDes Mode .....	793
14-462. E44h Internal Output A Trigger State Change Storage Mask .....	793
14-463. E48h Internal Output B Trigger State Change Storage Mask .....	793
14-464. EF0h Physical Layer Control .....	794
<b>Device-Specific Registers – Source Queue Weight</b>	
<b>and Soft Error (Offsets F00h – F3Ch) .....</b>	<b>796</b>
14-465. F00h ECRC Check Enable for Ports 0, 1, 2, 3 .....	797
14-466. F04h ECRC Check Enable for Ports 4, 5, 6, 7 .....	798
14-467. F08h ECRC Check Enable for Ports 8, 9, 10, 11 .....	799
14-468. F0Ch ECRC Check Enable for Ports 12, 13, 14, 15 .....	800
14-469. F10h Port Egress TLP Threshold .....	800
14-470. F18h Soft Error Counters 1 .....	801
14-471. F24h Soft Error Counters 4 .....	802
14-472. F28h Soft Error Counters 5 .....	803
14-473. F2Ch Soft Error Counters 6 .....	804
14-474. F30h Soft Error Injection .....	805
<b>Device-Specific Registers – Error Reporting</b>	
<b>(Offsets F40h – F4Ch) .....</b>	<b>806</b>
14-475. F40h Error Reporting Enable for Ports 0, 1, 2, 3 .....	807
14-476. F44h Error Reporting Enable for Ports 4, 5, 6, 7 .....	808
14-477. F48h Error Reporting Enable for Ports 8, 9, 10, 11 .....	809
14-478. F4Ch Error Reporting Enable for Ports 12, 13, 14, 15 .....	810
<b>Device-Specific Registers – ARI Capability</b>	
<b>(Offsets F50h – F8Ch) .....</b>	<b>811</b>
14-479. F50h Device Capability and Control 2 for Port 0 .....	812
14-480. F54h Device Capability and Control 2 for Port 1 .....	812
14-481. F58h Device Capability and Control 2 for Port 2 .....	813
14-482. F5Ch Device Capability and Control 2 for Port 3 .....	813
14-483. F60h Device Capability and Control 2 for Port 4 .....	814
14-484. F64h Device Capability and Control 2 for Port 5 .....	814
14-485. F68h Device Capability and Control 2 for Port 6 .....	815
14-486. F6Ch Device Capability and Control 2 for Port 7 .....	815
14-487. F70h Device Capability and Control 2 for Port 8 .....	816
14-488. F74h Device Capability and Control 2 for Port 9 .....	816
14-489. F78h Device Capability and Control 2 for Port 10 .....	817
14-490. F7Ch Device Capability and Control 2 for Port 11 .....	817

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

14-491. F80h Device Capability and Control 2 for Port 12 .....	818
14-492. F84h Device Capability and Control 2 for Port 13 .....	818
14-493. F88h Device Capability and Control 2 for Port 14 .....	819
14-494. F8Ch Device Capability and Control 2 for Port 15 .....	819
<b>Advanced Error Reporting Extended Capability</b>	
<b>Registers (Offsets FB4h – FDCh) .....</b>	<b>820</b>
14-495. FB4h Advanced Error Reporting Extended Capability Header .....	820
14-496. FB8h Uncorrectable Error Status .....	821
14-497. FBCh Uncorrectable Error Mask .....	822
14-498. FC0h Uncorrectable Error Severity .....	823
14-499. FC4h Correctable Error Status .....	825
14-500. FC8h Correctable Error Mask .....	826
14-501. FCCh Advanced Error Capabilities and Control .....	827
14-502. FD0h Header Log 0 .....	828
14-503. FD4h Header Log 1 .....	828
14-504. FD8h Header Log 2 .....	828
14-505. FDCh Header Log 3 .....	828

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**NT Port Virtual Interface Registers – NT Mode Only**

<b>NT Port Virtual Interface PCI-Compatible Type 0</b>	
<b>Configuration Header Registers (Offsets 00h – 3Ch) . . . . .</b>	<b>863</b>
16-1. 00h PCI Configuration ID . . . . .	863
16-2. 04h PCI Command/Status . . . . .	864
16-3. 08h PCI Class Code and Revision ID. . . . .	867
16-4. 0Ch Miscellaneous Control. . . . .	867
16-5. 10h Base Address 0 . . . . .	868
16-6. 14h Base Address 1 . . . . .	868
16-7. 18h Base Address 2 . . . . .	869
16-8. 1Ch Base Address 3. . . . .	869
16-9. 20h Base Address 4 . . . . .	870
16-10. 24h Base Address 5 . . . . .	870
16-11. 2Ch Subsystem ID and Subsystem Vendor ID. . . . .	871
16-12. 30h Expansion ROM Base Address . . . . .	871
16-13. 34h Capability Pointer. . . . .	872
16-14. 3Ch PCI Interrupt . . . . .	872
<b>NT Port Virtual Interface PCI Power Management</b>	
<b>Capability Registers (Offsets 40h – 44h) . . . . .</b>	<b>873</b>
16-15. 40h PCI Power Management Capability. . . . .	873
16-16. 44h PCI Power Management Status and Control. . . . .	874
<b>NT Port Virtual Interface MSI Capability Registers</b>	
<b>(Offsets 48h – 64h) . . . . .</b>	<b>876</b>
<b>NT Port Virtual Interface PCI Express</b>	
<b>Capability Registers (Offsets 68h – A0h) . . . . .</b>	<b>877</b>
16-17. 68h PCI Express Capability List and Capability . . . . .	878
16-18. 6Ch Device Capability . . . . .	879
16-19. 70h Device Status and Control. . . . .	880
16-20. 74h Link Capability . . . . .	882
16-21. 78h Link Status and Control . . . . .	885
16-22. 98h Link Status and Control 2 . . . . .	887
<b>NT Port Virtual Interface Subsystem ID and Subsystem</b>	
<b>Vendor ID Capability Registers (Offsets A4h – C4h) . . . . .</b>	<b>888</b>
16-23. A4h Subsystem Capability . . . . .	888
<b>NT Port Virtual Interface Vendor-Specific</b>	
<b>Capability 3 Registers (Offsets C8h – FCh) . . . . .</b>	<b>889</b>
16-24. C8h Vendor-Specific Capability 3 . . . . .	890
16-25. CCh Vendor-Specific Header 3. . . . .	890
16-26. D0h NT Port Virtual Interface BAR0/1 Setup . . . . .	891
16-27. D4h NT Port Virtual Interface Memory BAR2 Setup. . . . .	891
16-28. D8h NT Port Virtual Interface Memory BAR2/3 Setup . . . . .	892
16-29. DCh NT Port Virtual Interface Memory BAR4 Setup . . . . .	893
16-30. E0h NT Port Virtual Interface Memory BAR4/5 Setup . . . . .	894
16-31. F8h Configuration Address Window . . . . .	895
16-32. FCh Configuration Data Window . . . . .	895
<b>NT Port Virtual Interface Device Serial Number</b>	
<b>Extended Capability Registers (Offsets 100h – 134h) . . . . .</b>	<b>896</b>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

<b>NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh) . . . . .</b>	<b>897</b>
16-33. 148h Virtual Channel Extended Capability Header . . . . .	898
16-34. 160h VC0 Resource Status . . . . .	898
16-35. 16Ch VC1 Resource Status . . . . .	898
<b>NT Port Virtual Interface WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh) . . . . .</b>	<b>899</b>
<b>NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h) . . . . .</b>	<b>900</b>
<b>NT Port Virtual Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh) . . . . .</b>	<b>903</b>
16-36. 1DCh Debug Control . . . . .	904
16-37. 1E0h Power Management Hot Plug User Configuration . . . . .	913
16-38. 1E8h Bad TLP Counter . . . . .	913
16-39. 1ECh Bad DLLP Counter . . . . .	913
16-40. 1F8h ACK Transmission Latency Limit . . . . .	914
<b>NT Port Virtual Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch) . . . . .</b>	<b>915</b>
16-41. 660h Ingress Control . . . . .	915
16-42. 664h Ingress Control Shadow . . . . .	916
<b>NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch) . . . . .</b>	<b>917</b>
16-43. 950h Vendor-Specific Extended Capability 2 . . . . .	917
<b>NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h) . . . . .</b>	<b>918</b>
16-44. C34h Vendor-Specific Extended Capability 4 . . . . .	919
16-45. C38h Vendor-Specific Header 4 . . . . .	919
16-46. C3Ch Memory BAR2 Address Translation Lower . . . . .	920
16-47. C40h Memory BAR3 Address Translation Upper . . . . .	920
16-48. C44h Memory BAR4 Address Translation Lower . . . . .	920
16-49. C48h Memory BAR5 Address Translation Upper . . . . .	920
16-50. C4Ch Virtual Interface IRQ Set . . . . .	921
16-51. C50h Virtual Interface IRQ Clear . . . . .	921
16-52. C54h Virtual Interface IRQ Mask Set . . . . .	922
16-53. C58h Virtual Interface IRQ Mask Clear . . . . .	922
16-54. C5Ch Link Interface IRQ Set . . . . .	923
16-55. C60h Link Interface IRQ Clear . . . . .	923
16-56. C64h Link Interface IRQ Mask Set . . . . .	924
16-57. C68h Link Interface IRQ Mask Clear . . . . .	924
16-58. C6Ch NT Port SCRATCH0 . . . . .	925
16-59. C70h NT Port SCRATCH1 . . . . .	925
16-60. C74h NT Port SCRATCH2 . . . . .	925
16-61. C78h NT Port SCRATCH3 . . . . .	925
16-62. C7Ch NT Port SCRATCH4 . . . . .	925
16-63. C80h NT Port SCRATCH5 . . . . .	926
16-64. C84h NT Port SCRATCH6 . . . . .	926
16-65. C88h NT Port SCRATCH7 . . . . .	926
<b>NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – EECh) . . . . .</b>	<b>927</b>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>NT Port Virtual Interface NT Bridging-Specific Registers – Base Address and Base Address Setup (Offsets D68h – D90h) . . . . .</b>	<b>928</b>
16-66. D68h BAR0 (Shadow Copy) . . . . .	929
16-67. D6Ch BAR1 (Shadow Copy) . . . . .	929
16-68. D70h BAR2 (Shadow Copy) . . . . .	930
16-69. D74h BAR3 (Shadow Copy) . . . . .	930
16-70. D78h BAR4 (Shadow Copy) . . . . .	931
16-71. D7Ch BAR5 (Shadow Copy) . . . . .	931
16-72. D80h BAR0/1 Setup (Shadow Copy) . . . . .	932
16-73. D84h Memory BAR2 Setup (Shadow Copy) . . . . .	932
16-74. D88h Memory BAR2/3 Setup (Shadow Copy) . . . . .	933
16-75. D8Ch Memory BAR4 Setup (Shadow Copy) . . . . .	934
16-76. D90h Memory BAR4/5 Setup (Shadow Copy) . . . . .	935
<b>NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DB0h) . . . . .</b>	<b>936</b>
16-77. D94h – DB0h NT Port Virtual Interface Requester ID Translation LUT Entry_n . . . . .	937
<b>NT Port Virtual Interface Device-Specific Registers (Offsets E40h – F8Ch) . . . . .</b>	<b>938</b>
<b>NT Port Virtual Interface Device-Specific Registers – Physical Layer (Offsets E40h – EFCh) . . . . .</b>	<b>939</b>
<b>NT Port Virtual Interface Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch) . . . . .</b>	<b>939</b>
<b>NT Port Virtual Interface Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch) . . . . .</b>	<b>940</b>
16-78. F50h Device Capability and Control 2 for Port 0 . . . . .	941
16-79. F54h Device Capability and Control 2 for Port 1 . . . . .	941
16-80. F58h Device Capability and Control 2 for Port 2 . . . . .	942
16-81. F5Ch Device Capability and Control 2 for Port 3 . . . . .	942
16-82. F60h Device Capability and Control 2 for Port 4 . . . . .	943
16-83. F64h Device Capability and Control 2 for Port 5 . . . . .	943
16-84. F68h Device Capability and Control 2 for Port 6 . . . . .	944
16-85. F6Ch Device Capability and Control 2 for Port 7 . . . . .	944
16-86. F70h Device Capability and Control 2 for Port 8 . . . . .	945
16-87. F74h Device Capability and Control 2 for Port 9 . . . . .	945
16-88. F78h Device Capability and Control 2 for Port 10 . . . . .	946
16-89. F7Ch Device Capability and Control 2 for Port 11 . . . . .	946
16-90. F80h Device Capability and Control 2 for Port 12 . . . . .	947
16-91. F84h Device Capability and Control 2 for Port 13 . . . . .	947
16-92. F88h Device Capability and Control 2 for Port 14 . . . . .	948
16-93. F8Ch Device Capability and Control 2 for Port 15 . . . . .	948
<b>NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .</b>	<b>949</b>
16-94. FB4h Advanced Error Reporting Extended Capability Header . . . . .	949
<b>NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh) . . . . .</b>	<b>950</b>
16-95. FE0h Link Error Status Virtual . . . . .	951
16-96. FE4h Link Error Mask Virtual . . . . .	951

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**NT Port Link Interface Registers – NT Mode Only**

<b>NT Port Link Interface PCI-Compatible Type 0</b>	
<b>Configuration Header Registers (Offsets 00h – 3Ch) .....</b>	<b>961</b>
17-1. 00h PCI Configuration ID .....	961
17-2. 04h PCI Command/Status .....	962
17-3. 08h PCI Class Code and Revision ID .....	964
17-4. 0Ch Miscellaneous Control .....	965
17-5. 10h Base Address 0 .....	966
17-6. 14h Base Address 1 .....	966
17-7. 18h Base Address 2 .....	967
17-8. 1Ch Base Address 3 .....	967
17-9. 20h Base Address 4 .....	968
17-10. 24h Base Address 5 .....	968
17-11. 2Ch Subsystem ID and Subsystem Vendor ID .....	969
17-12. 30h Expansion ROM Base Address .....	969
17-13. 34h Capability Pointer .....	970
17-14. 3Ch PCI Interrupt .....	970
<b>NT Port Link Interface PCI Power Management</b>	
<b>Capability Registers (Offsets 40h – 44h) .....</b>	<b>971</b>
17-15. 40h PCI Power Management Capability .....	971
17-16. 44h PCI Power Management Status and Control .....	972
<b>NT Port Link Interface MSI Capability Registers</b>	
<b>(Offsets 48h – 64h) .....</b>	<b>974</b>
17-17. 48h MSI Capability .....	975
17-18. 58h MSI Mask .....	976
17-19. 5Ch MSI Status .....	977
<b>NT Port Link Interface PCI Express Capability</b>	
<b>Registers (Offsets 68h – A0h) .....</b>	<b>978</b>
17-20. 68h PCI Express Capability List and Capability .....	978
17-21. 6Ch Device Capability .....	979
17-22. 70h Device Status and Control .....	980
17-23. 74h Link Capability .....	982
17-24. 78h Link Status and Control .....	984
17-25. 98h Link Status and Control 2 .....	986
<b>NT Port Link Interface Subsystem ID and Subsystem</b>	
<b>Vendor ID Capability Registers (Offsets A4h – C4h) .....</b>	<b>987</b>
17-26. A4h Subsystem Capability .....	987
<b>NT Port Link Interface Vendor-Specific Capability 3</b>	
<b>Registers (Offsets C8h – FCh) .....</b>	<b>988</b>
17-27. C8h Vendor-Specific Extended Capability 3 .....	989
17-28. CCCh Vendor-Specific Header 3 .....	989
17-29. E4h NT Port Link Interface BAR0/1 Setup .....	990
17-30. E8h NT Port Link Interface Memory BAR2 Setup .....	990
17-31. ECCh NT Port Link Interface Memory BAR2/3 Setup .....	991
17-32. F0h NT Port Link Interface Memory BAR4 Setup .....	992
17-33. F4h NT Port Link Interface Memory BAR4/5 Setup .....	993
17-34. F8h Configuration Address Window .....	994
17-35. FCh Configuration Data Window .....	994

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

<b>NT Port Link Interface Device Serial Number</b>	
<b>Extended Capability Registers (Offsets 100h – 134h)</b>	995
<b>NT Port Link Interface Power Budget</b>	
<b>Extended Capability Registers (Offsets 138h – 144h)</b>	995
<b>NT Port Link Interface Virtual Channel</b>	
<b>Extended Capability Registers (Offsets 148h – 1BCh)</b>	996
17-36. 148h Virtual Channel Extended Capability Header	996
<b>NT Port Link Interface Device-Specific Registers</b>	
<b>(Offsets 1C0h – C88h)</b>	997
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Error Checking and Debug (Offsets 1C0h – 1FCh)</b>	998
17-37. 1C8h ECC Error Check Disable	999
17-38. 1E0h Power Management Hot Plug User Configuration	1000
17-39. 1F8h ACK Transmission Latency Limit	1001
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Ingress Control and Port Enable (Offsets 660h – 67Ch)</b>	1002
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)</b>	1003
17-40. 950h Vendor-Specific Extended Capability 2	1003
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Ingress NT Control (Offset 960h)</b>	1004
17-41. 960h Ingress NT Control 0	1004
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)</b>	1005
17-42. C3Ch Memory BAR2 Address Translation Lower	1006
17-43. C40h Memory BAR3 Address Translation Upper	1006
17-44. C44h Memory BAR4 Address Translation Lower	1006
17-45. C48h Memory BAR5 Address Translation Upper	1006
<b>NT Port Link Interface NT Bridging-Specific Registers</b>	
<b>(Offsets C8Ch – EFCh)</b>	1007
<b>NT Port Link Interface Bridging-Specific Registers –</b>	
<b>Requester ID Translation Lookup Table Entry</b>	
<b>(Addresses DB4h – DF0h)</b>	1008
17-46. DB4h – DF0h NT Port Link Interface Requester ID Translation LUT Entry_n_m	1009
<b>NT Port Link Interface Device-Specific Registers –</b>	
<b>Source Queue Weight (Offsets F00h – F3Ch)</b>	1010
<b>NT Port Link Interface Advanced Error Reporting</b>	
<b>Extended Capability Registers (Offsets FB4h – FDCh)</b>	1011
17-47. FB4h Advanced Error Reporting Extended Capability Header	1011

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**DMA Interface Registers – DMA Mode Only**

<b>DMA Interface PCI-Compatible Type 0 Configuration</b>	
<b>Header Registers (Offsets 00h – 3Ch) .....</b>	<b>1020</b>
18-1. 00h PCI Configuration ID.....	1020
18-2. 04h PCI Command/Status.....	1021
18-3. 08h DMA PCI Class Code and Revision ID.....	1024
18-4. 0Ch DMA Miscellaneous Control.....	1025
18-5. 10h Base Address 0 DMA.....	1026
18-6. 14h Base Address 1 DMA.....	1026
18-7. 2Ch Subsystem ID and Subsystem Vendor ID.....	1027
18-8. 30h Expansion ROM Base Address.....	1027
18-9. 34h Capability Pointer .....	1027
18-10. 3Ch PCI Interrupt.....	1028
<b>DMA Interface PCI Power Management Capability</b>	
<b>Registers (Offsets 40h – 44h) .....</b>	<b>1029</b>
18-11. 40h PCI Power Management Capability .....	1029
18-12. 44h PCI Power Management Status and Control .....	1030
<b>DMA Interface MSI Capability Registers</b>	
<b>(Offsets 48h – 64h) .....</b>	<b>1032</b>
18-13. 48h DMA MSI Capability.....	1033
18-14. 4Ch DMA MSI Address.....	1033
18-15. 50h DMA MSI Upper Address.....	1034
18-16. 54h DMA MSI Data .....	1034
18-17. 58h DMA MSI Mask.....	1035
18-18. 5Ch DMA MSI Status .....	1035
<b>DMA Interface PCI Express Capability Registers</b>	
<b>(Offsets 68h – A0h).....</b>	<b>1036</b>
18-19. 68h PCI Express Capability List and Capability .....	1036
18-20. 6Ch Device Capability .....	1037
18-21. 70h Device Status and Control .....	1038
18-22. 74h Link Capability .....	1040
18-23. 78h Link Status and Control .....	1043
18-24. 8Ch Device Capability 2 .....	1044
18-25. 90h Device Status and Control 2 .....	1045
18-26. 98h Link Status and Control 2 .....	1045
<b>DMA Interface Subsystem ID and Subsystem</b>	
<b>Vendor ID Capability Registers (Offsets A4h – C4h).....</b>	<b>1046</b>
18-27. A4h Subsystem Capability.....	1046
18-28. A8h Subsystem ID and Subsystem Vendor ID .....	1046
<b>DMA Interface Base Address DMA Setup Register</b>	
<b>(Offsets C8h – FCh) .....</b>	<b>1047</b>
18-29. E4h BAR0/1 DMA Setup .....	1047
<b>DMA Interface Device Serial Number</b>	
<b>Extended Capability Registers (Offsets 100h – 134h).....</b>	<b>1048</b>
18-30. 100h Device Serial Number Extended Capability Header.....	1048
18-31. 104h Serial Number (Lower DW).....	1049
18-32. 108h Serial Number (Upper DW).....	1049

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved – PLX CONFIDENTIAL - NDA REQUIRED

<b>Vendor-Specific Extended Capability Registers –</b>	
<b>DMA Channel (Offsets 1F4h – 5FCh) . . . . .</b>	<b>1050</b>
18-33. 1F4h Vendor-Specific Extended Capability Header . . . . .	1054
18-34. 1F8h Vendor-Specific Header 1 . . . . .	1054
18-35. 1FCh DMA Global Control . . . . .	1054
18-36. 200h DMA Channel 0 Source Address Lower . . . . .	1055
18-37. 204h DMA Channel 0 Source Address Upper . . . . .	1055
18-38. 208h DMA Channel 0 Destination Address Lower . . . . .	1055
18-39. 20Ch DMA Channel 0 Destination Address Upper . . . . .	1055
18-40. 210h DMA Channel 0 Transfer Size . . . . .	1056
18-41. 214h DMA Channel 0 Descriptor Ring Address Lower . . . . .	1056
18-42. 218h DMA Channel 0 Descriptor Ring Address Upper . . . . .	1056
18-43. 21Ch DMA Channel 0 Next Descriptor Ring Address Lower . . . . .	1056
18-44. 220h DMA Channel 0 Descriptor Ring Size . . . . .	1057
18-45. 224h DMA Channel 0 Last Descriptor Address Lower . . . . .	1057
18-46. 228h DMA Channel 0 Last Descriptor Transfer Size . . . . .	1057
18-47. 22Ch DMA Channel 0 Bandwidth . . . . .	1057
18-48. 234h DMA Channel 0 Maximum Prefetch Limit . . . . .	1058
18-49. 238h DMA Channel 0 Control/Status . . . . .	1059
18-50. 23Ch DMA Channel 0 Interrupt Control/Status . . . . .	1061
18-51. 240h DMA Channel 0 Error Header Log 0 . . . . .	1062
18-52. 244h DMA Channel 0 Error Header Log 1 . . . . .	1062
18-53. 248h DMA Channel 0 Error Header Log 2 . . . . .	1062
18-54. 24Ch DMA Channel 0 Error Header Log 3 . . . . .	1062
18-55. 250h DMA Channel 0 Device-Specific Error Status . . . . .	1063
18-56. 254h DMA Channel 0 Maximum Outstanding Read/Write Request . . . . .	1063
18-57. 300h DMA Channel 1 Source Address Lower . . . . .	1064
18-58. 304h DMA Channel 1 Source Address Upper . . . . .	1064
18-59. 308h DMA Channel 1 Destination Address Lower . . . . .	1064
18-60. 30Ch DMA Channel 1 Destination Address Upper . . . . .	1064
18-61. 310h DMA Channel 1 Transfer Size . . . . .	1065
18-62. 314h DMA Channel 1 Descriptor Ring Address Lower . . . . .	1065
18-63. 318h DMA Channel 1 Descriptor Ring Address Upper . . . . .	1065
18-64. 31Ch DMA Channel 1 Next Descriptor Ring Address Lower . . . . .	1065
18-65. 320h DMA Channel 1 Descriptor Ring Size . . . . .	1066
18-66. 324h DMA Channel 1 Last Descriptor Address Lower . . . . .	1066
18-67. 328h DMA Channel 1 Last Descriptor Transfer Size . . . . .	1066
18-68. 32Ch DMA Channel 1 Bandwidth . . . . .	1066
18-69. 334h DMA Channel 1 Maximum Prefetch Limit . . . . .	1067
18-70. 338h DMA Channel 1 Control/Status . . . . .	1068
18-71. 33Ch DMA Channel 1 Interrupt Control/Status . . . . .	1070
18-72. 340h DMA Channel 1 Error Header Log 0 . . . . .	1071
18-73. 344h DMA Channel 1 Error Header Log 1 . . . . .	1071
18-74. 348h DMA Channel 1 Error Header Log 2 . . . . .	1071
18-75. 34Ch DMA Channel 1 Error Header Log 3 . . . . .	1071
18-76. 350h DMA Channel 1 Device-Specific Error Status . . . . .	1072
18-77. 354h DMA Channel 1 Maximum Outstanding Read/Write Request . . . . .	1072
18-78. 400h DMA Channel 2 Source Address Lower . . . . .	1073
18-79. 404h DMA Channel 2 Source Address Upper . . . . .	1073
18-80. 408h DMA Channel 2 Destination Address Lower . . . . .	1073
18-81. 40Ch DMA Channel 2 Destination Address Upper . . . . .	1073
18-82. 410h DMA Channel 2 Transfer Size . . . . .	1074
18-83. 414h DMA Channel 2 Descriptor Ring Address Lower . . . . .	1074
18-84. 418h DMA Channel 2 Descriptor Ring Address Upper . . . . .	1074
18-85. 41Ch DMA Channel 2 Next Descriptor Ring Address Lower . . . . .	1074

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

18-86. 420h DMA Channel 2 Descriptor Ring Size .....	1075
18-87. 424h DMA Channel 2 Last Descriptor Address Lower .....	1075
18-88. 428h DMA Channel 2 Last Descriptor Transfer Size .....	1075
18-89. 42Ch DMA Channel 2 Bandwidth .....	1075
18-90. 434h DMA Channel 2 Maximum Prefetch Limit .....	1076
18-91. 438h DMA Channel 2 Control/Status .....	1077
18-92. 43Ch DMA Channel 2 Interrupt Control/Status .....	1079
18-93. 440h DMA Channel 2 Error Header Log 0 .....	1080
18-94. 444h DMA Channel 2 Error Header Log 1 .....	1080
18-95. 448h DMA Channel 2 Error Header Log 2 .....	1080
18-96. 44Ch DMA Channel 2 Error Header Log 3 .....	1080
18-97. 450h DMA Channel 2 Device-Specific Error Status .....	1081
18-98. 454h DMA Channel 2 Maximum Outstanding Read/Write Request .....	1081
18-99. 500h DMA Channel 3 Source Address Lower .....	1082
18-100. 504h DMA Channel 3 Source Address Upper .....	1082
18-101. 508h DMA Channel 3 Destination Address Lower .....	1082
18-102. 50Ch DMA Channel 3 Destination Address Upper .....	1082
18-103. 510h DMA Channel 3 Transfer Size .....	1083
18-104. 514h DMA Channel 3 Descriptor Ring Address Lower .....	1083
18-105. 518h DMA Channel 3 Descriptor Ring Address Upper .....	1083
18-106. 51Ch DMA Channel 3 Next Descriptor Ring Address Lower .....	1083
18-107. 520h DMA Channel 3 Descriptor Ring Size .....	1084
18-108. 524h DMA Channel 3 Last Descriptor Address Lower .....	1084
18-109. 528h DMA Channel 3 Last Descriptor Transfer Size .....	1084
18-110. 52Ch DMA Channel 3 Bandwidth .....	1084
18-111. 534h DMA Channel 3 Maximum Prefetch Limit .....	1085
18-112. 538h DMA Channel 3 Control/Status .....	1086
18-113. 53Ch DMA Channel 3 Interrupt Control/Status .....	1088
18-114. 540h DMA Channel 3 Error Header Log 0 .....	1089
18-115. 544h DMA Channel 3 Error Header Log 1 .....	1089
18-116. 548h DMA Channel 3 Error Header Log 2 .....	1089
18-117. 54Ch DMA Channel 3 Error Header Log 3 .....	1089
18-118. 550h DMA Channel 3 Device-Specific Error Status .....	1090
18-119. 554h DMA Channel 3 Maximum Outstanding Read/Write Request .....	1090
<b>DMA Device-Specific Registers – RAM Test and Soft Error (Offsets 600h – FB0h) .....</b>	<b>1091</b>
18-120. 600h RAM Test Control .....	1091
18-121. 604h Soft Error Injection Control/Status .....	1092
<b>DMA Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) .....</b>	<b>1093</b>
18-122. FB4h Advanced Error Reporting Extended Capability Header .....	1093
18-123. FB8h Uncorrectable Error Status .....	1094
18-124. FBCh Uncorrectable Error Mask .....	1095
18-125. FC0h Uncorrectable Error Severity .....	1096
18-126. FC4h Correctable Error Status .....	1097
18-127. FC8h Correctable Error Mask .....	1098
18-128. FCCh Advanced Error Capabilities and Control .....	1099
18-129. FD0h Header Log 0 .....	1100
18-130. FD4h Header Log 1 .....	1100
18-131. FD8h Header Log 2 .....	1100
18-132. FDCh Header Log 3 .....	1100

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

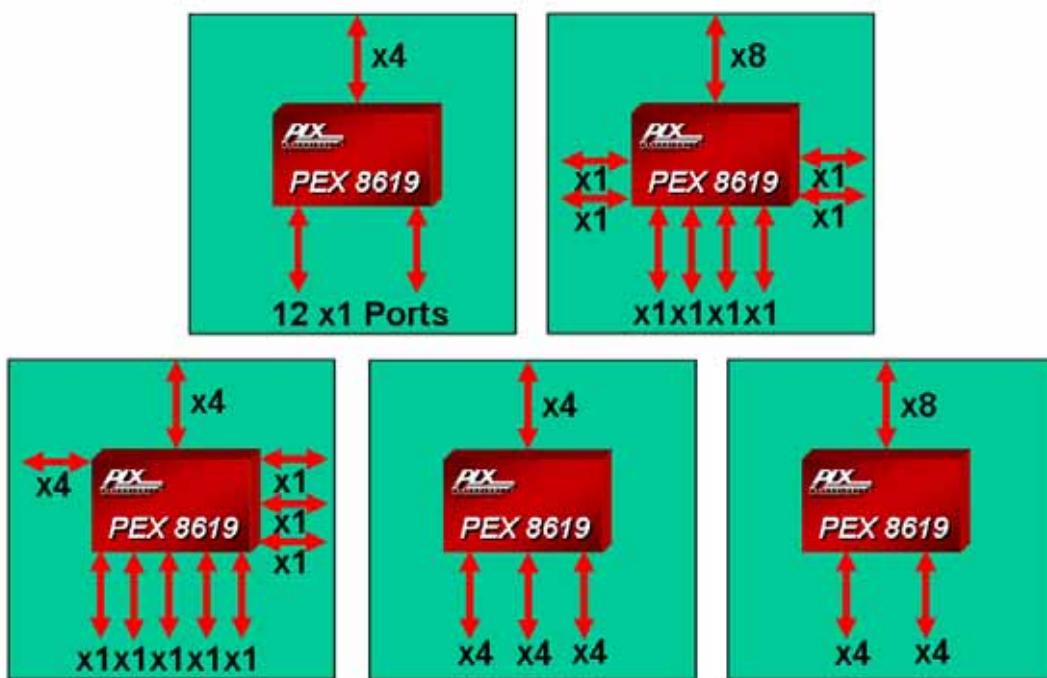
ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 1.1 Overview

This data book describes PLX Technology's ExpressLane™ PEX 8619, a fully non-blocking, low-latency, low-cost, and low-power 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA. Conforming to the *PCI Express Base r2.0*, the PEX 8619 enables users to add scalable, high-bandwidth I/O to various products, including communications platforms, servers, storage systems, embedded systems, and intelligent I/O modules. The PEX 8619's flexible hardware configuration and software programmability allows the switch to be tailored for a wide variety of applications, including fan-out, aggregation, and peer-to-peer applications.

The PEX 8619 supports multiple Port configuration options, to provide flexible solutions for optimal product design. Figure 1-1 illustrates a subset of the Port configurations supported by the PEX 8619. The PEX 8619 can also support x2 Ports, by auto-negotiating its Ports to the Link width of the PCI Express device with which it is interfacing.

**Figure 1-1. Port Configuration Examples**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 1.2 Features

The PEX 8619 supports the following features:

- 16-Port PCI Express switch
  - 16 Lanes with integrated on-chip SerDes
  - Low-power SerDes (under 90 mW per Lane)
  - Port configuration
    - 16 independent Ports
    - Choice of Link width (quantity of Lanes) per unique Link/Port – x1, x4, or x8; x2 is also supported
    - Configurable with Strapping balls, serial EEPROM, I<sup>2</sup>C, SMBus, and/or Host software
    - Designate any Port as the *upstream Port* (Port 0 is recommended)
- High Performance
  - 160 GT/s aggregate bandwidth (5.0 GT/s/Lane x 16 SerDes x 2 (full duplex))
  - Integrated 5.0 GT/s SerDes speed negotiation, for each Port
  - Full line rate on all Ports
  - Cut-Thru latency – 140 ns for Link widths of x4 to x1
  - Non-blocking Crossbar Switch interface supports TLP bandwidth capacity of each x8 Link
  - Maximum Payload Size – 2,048 bytes
- *performancePAK*<sup>TM</sup>
  - Read Pacing<sup>TM</sup> (intelligent bandwidth allocation)
  - Dual Cast<sup>TM</sup>
  - Dynamic Buffer Pool Architecture for faster credit updates
  - Integrated Direct Memory Access (DMA) engine
  - Four DMA channels
    - Internal Descriptor support
    - DMA function independent from Transparent switch function
- *visionPAK*<sup>TM</sup>
  - Performance Monitoring
    - Per-Port Payload and Header Counters
    - Per-traffic type (Write, Read, Completion) Counters
  - PCI Express Packet Generator
    - Capable of saturating a x8 Gen 2 Link
  - Error Injection and Pseudo-Random Bit Sequence (PRBS)
  - SerDes Eye Capture
  - SerDes Loopback
- 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
  - Two Virtual Channels (VC0 and VC1), per Port
  - Eight Traffic Classes (TC[7:0]), per Port
  - Weighted Round-Robin (WRR) Port and Virtual Channel (VC) arbitration

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

- Non-Transparent Bridging
  - Configurable Non-Transparent Port
  - Moveable *upstream Port*
  - Enables Dual-Host, Dual-Fabric, Host-Failover applications
  - Cross-link Port capability
- Reliability, Availability, Serviceability (RAS) features
  - All Transparent downstream Ports are Hot Plug-capable, through I<sup>2</sup>C (Serial Hot Plug Controller on every Port)
  - End-to-end Cyclic Redundancy Check (ECRC) and Poison bit support
  - Data path protection
  - Memory (RAM) error correction
  - Advanced Error Reporting support
  - Lane Status bits and GPIO available
  - Per-Port error diagnostics
  - Joint Test Action Group (JTAG) AC/DC boundary scan
- Spread-Spectrum Clock isolation
  - Dual-clock domain
- INTA# ([PEX\\_INTA#](#)) and FATAL ERROR ([FATAL\\_ERR#](#))  
(Conventional PCI SERR# equivalent) ball support
- 32 General-Purpose Input/Output (GPIO) balls, which can be used for GPIO and/or Interrupt inputs
- Lane Status balls ([PEX\\_LANE\\_GOOD\[15:0\]#](#))
- 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 or 5.0 GT/s) 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 – JTAG support for DC
- 19 x 19 mm<sup>2</sup>, 324-ball Heat Slug Ball Grid Array (HSBGA) package
- Typical power – 1.70W (Gen 1, 2.5 GT/s) or 1.99W (Gen 2, 5.0 GT/s)
- Microsoft® Windows® Logo (WHQL)-compliant

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

- 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 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 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)*
  - *System Management Bus (SMBus) Specification, Version 2.0 (SMBus v2.0)*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 2 Features and Applications

### 2.1 Flexible and Feature-Rich 16-Lane, 16-Port Switch

#### 2.1.1 Highly Flexible Port Configurations

The PLX ExpressLane PEX 8619 PCI Express Gen 2 Switch with DMA offers a maximum of 16 configurable Ports. To support specific bandwidth needs, Link widths can be individually configured as any power-of-two, from x1 to x8, through auto-negotiation, hardware strapping, Host software configuration, an optional serial EEPROM, and/or the I<sup>2</sup>C Slave interface. Flexible buffer allocation, along with the PEX 8619's **flexible packet flow control**, maximizes throughput for applications where more traffic flows in the downstream, rather than upstream, direction.

The PEX 8619 supports several Port configurations. *For example*, the PEX 8619 can be used in a fan-out application, where any one Port is the upstream Port and the remaining available Lanes are divided among up to 15 downstream Ports, of varying Link widths. Any one Port can be designated as, or dynamically changed to be, the upstream Port (Port 0 is recommended).

#### 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). The Crossbar Switch protocol is sufficiently flexible and robust to support a variety of embedded system requirements. The Crossbar Switch architecture basic features include:

- Multiple concurrent Data transfers
- Global ordering within the PEX 8619
- 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 2.1.3 Low Packet Latency and High Performance

The PEX 8619 architecture supports packet Cut-Thru with a maximum latency of 140 ns for Link widths of x4 to x1. This, combined with large Packet memory and Non-Blocking Internal Switch architecture, provides full line rate on all Ports for low-latency applications, *such as* communications and servers. The low latency enables applications to achieve high throughput and performance. In addition to low latency, the PEX 8619 supports a 2,048-byte **Maximum Payload Size**, enabling users to achieve even higher throughput.

### 2.1.3.1 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). 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.

### 2.1.3.2 Cut-Thru

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 8619 is designed to cut through TLPs, to and from every Port. By default, all Ports are enabled for Cut-Thru. Cut-Thru mode can be disabled for all Ports, by Clearing the **Debug Control** register **Cut-Thru Enable** bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[21]**).

Cut-Thru mode, if enabled, is supported for the PEX 8619's NT Port Link Interface, if the PEX 8619 is configured for NT mode.

**Note:** The **Debug Control** register **Cut-Thru Enable** bit affects the entire switch. If Cut-Thru is enabled, all Ports use Cut-Thru. If Cut-Thru is **not** enabled, no Ports use Cut-Thru.

**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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 2.1.4 Virtual Channels and Traffic Classes

The PEX 8619 supports two Virtual Channels (VC0 and VC1) and eight Traffic Classes (TC[7:0]). VC0 and TC0 are required by the *PCI Express Base r2.0*, and configured at device start-up.

For further details, refer to [Section 8.2, “Virtual Channel Support.”](#)

## 2.1.5 Port Arbitration

The PEX 8619 supports hardware-fixed and Weighted Round-Robin (WRR) Ingress Port Arbitration. This allows fine-tuning of Quality of Service (QoS) and efficient use of Packet buffers, for better system performance. The PEX 8619 also supports:

- RR VC arbitration on the ingress Ports
- WRR VC arbitration on the egress Ports, that select between VC0 and VC1, for each egress Port

For further details, refer to [Section 8.2, “Virtual Channel Support.”](#)

## 2.1.6 End-to-End Packet Integrity

To enable designs that require **guaranteed error-free packets**, the PEX 8619 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 8619, by providing automatic correction of any 1-bit errors. These features are optional in the *PCI Express Base r2.0*; however, PLX provides them across its entire ExpressLane PCI Express Gen 2 switch product line.

## 2.1.7 Configuration Flexibility

The PEX 8619 provides several ways to configure its operations. *For example*, the PEX 8619 can be configured through Strapping balls, CPU Configuration Requests, 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 2.1.8 Interoperability

The PEX 8619 is designed to be fully compliant with the *PCI Express Base r2.0*, and is backward-compatible to the *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. 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 8619 provides **low-power** capability that is fully compliant with the *PCI Express Base r2.0* and *PCI Power Mgmt. r1.2* Power Management (PM) specifications. Unused SerDes are automatically powered down, to further reduce power consumption.

The PEX 8619 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 8619 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 crisscross 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 8619 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 2.1.11 **performancePAK**

Exclusive to PLX, *performancePAK* is a suite of unique and innovative performance features that enable PLX's Gen 2 switches to be the highest-performing Gen 2 switches available in the market today. The *performancePAK* features consist of Read Pacing, Dual Cast, Dynamic Buffer Pool, and Integrated Direct Memory Access Engine.

### 2.1.11.1 **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 9.6, "Read Pacing."](#)

### 2.1.11.2 **Dual Cast**

The Dual Cast feature allows for the copying of data (packets) from one ingress Port to two egress Ports, in a single transaction, allowing for higher performance in storage, security, and mirroring applications. The feature relieves the CPU from having to conduct two separate transactions, resulting in higher system performance.

For further details, refer to [Section 9.7, "Dual Cast."](#)

### 2.1.11.3 **Dynamic Buffer Pool**

The PEX 8619 uses a dynamic buffer pool for FC management, which uses a common pool of FC Credits that is shared among other Ports within a Half-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 Half-Station, for faster FC credit updates.

For further details, refer to [Section 9.3.2, "Dynamic Buffering."](#)

### 2.1.11.4 **Integrated Direct Memory Access Engine**

The PEX 8619 implements an internal Direct Memory Access (DMA) engine that can be programmed to move data between two address locations. The DMA engine consists of four channels, each capable of saturating a PCI Express x8 Gen 2 Link in one direction. Once programmed, the PEX 8619's DMA engine moves the data between the source and destination. Optionally, the DMA engine can be programmed to generate an event to the Host after the DMA transfer completes.

For further details, refer to [Chapter 10, "Integrated DMA Engine – DMA Mode Only."](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 2.1.12 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.

### 2.1.12.1 Performance Monitoring

The PEX 8619'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).

### 2.1.12.2 Error Injection

Using the PEX 8619'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.

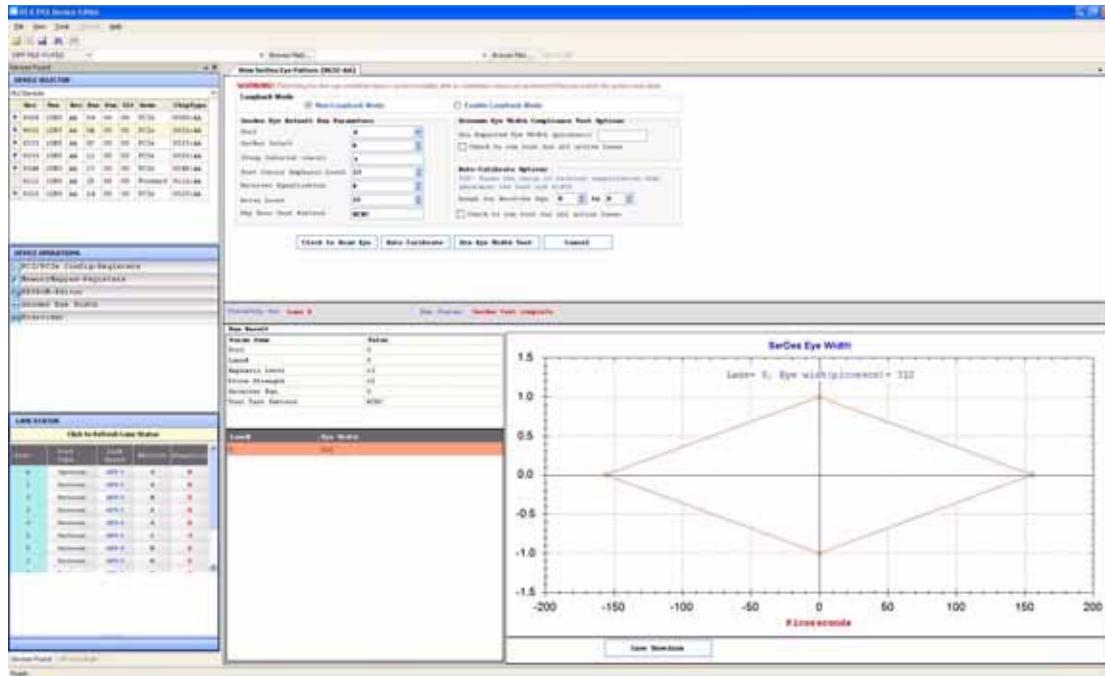
### 2.1.12.3 SerDes Loopback

The PEX 8619 supports External Tx, Recovered Clock, and Recovered Data Loopback modes.

### 2.1.12.4 SerDes Eye Capture

Users can evaluate their system's signal integrity at the Physical Layer (PHY), using the PEX 8619's SerDes Eye Capture feature. Using PLX's SDK, users can view the Receiver eye width of any Lane on the PEX 8619. Users can then modify SerDes Settings and see the impact on the Receiver eye. Figure 2-1 presents a screen shot of the SDK's SerDes Eye Capture feature.

**Figure 2-1. PLX SDK SerDes Eye Capture Feature**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 2.1.13 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 8619 Hot Plug Capability and Advanced Error Reporting features make the switch suitable for **High-Availability (HA) applications**. If the PEX 8619 is used in an application where one or more of its downstream Ports connect to PCI Express slots, each Port's Serial Hot Plug Controller can be used to manage the Hot Plug event of its associated slot. Each PEX 8619 Transparent downstream Port is equipped with a **Hot Plug Control/Status** register, to support Hot Plug capability through external logic (by way of the I<sup>2</sup>C Master interface).

For further details, refer to [Chapter 12, “Hot Plug Support.”](#)

## 2.1.14 Fully Compliant Power Management

The PEX 8619 supports Link (L0, L0s, L1, L2/L3 Ready, and L3) and Device (D0 and D3hot) PM states, in compliance with the *PCI Express Base r2.0* and *PCI Power Mgmt. r1.2* PM specifications.

For further details, refer to [Chapter 13, “Power Management.”](#)

## 2.1.15 General-Purpose Input/Output Signals

The PEX 8619 contains 32 General-Purpose Input/Output (GPIO) balls and associated registers, that can be programmed to function as GPIO and/or Interrupt inputs. Default functionality is GPIO inputs; however, serial EEPROM, I<sup>2</sup>C/SMBus, and/or software can program the **GPIO** registers to define functionality for each I/O.

For further details, refer to the [GPIO\[31:0\]](#) signal descriptions in [Section 3.4.7, “Device-Specific Signals,”](#) and [Section 11.5, “General-Purpose Input/Output.”](#)

## 2.1.16 Non-Transparent Dual Host and Failover Support – NT Mode Only

The PEX 8619 supports full Non-Transparent Bridging (NTB) functionality, to allow implementation of **multi-Host systems** and **intelligent I/O modules** in applications that require redundancy support, *such as communications, storage, and servers.*

NT bridges allow systems to isolate Host memory domains, by presenting the processor subsystem as an endpoint, rather than 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

For further details, refer to [Chapter 15, “Non-Transparent Bridging – NT Mode Only.”](#)

## 2.1.17 Spread-Spectrum Clock Isolation

The PEX 8619 supports Dual Clock domains. This feature allows the buffer inside the PEX 8619 to become part of two systems with different system clocks, one of which can implement Spread-Spectrum Clocking. The constant clock input must follow the clock requirements defined in the *PCI Express Base r2.0* ( $\pm 300$  ppm).

For further details, refer to [Chapter 19, “Dual Clocking Support.”](#)

## 2.2 Applications

Suitable for **fan-out**, **control plane applications**, and **embedded systems**, as well as **intelligent I/O applications**, the PEX 8619 can be configured for a wide variety of form factors and applications.

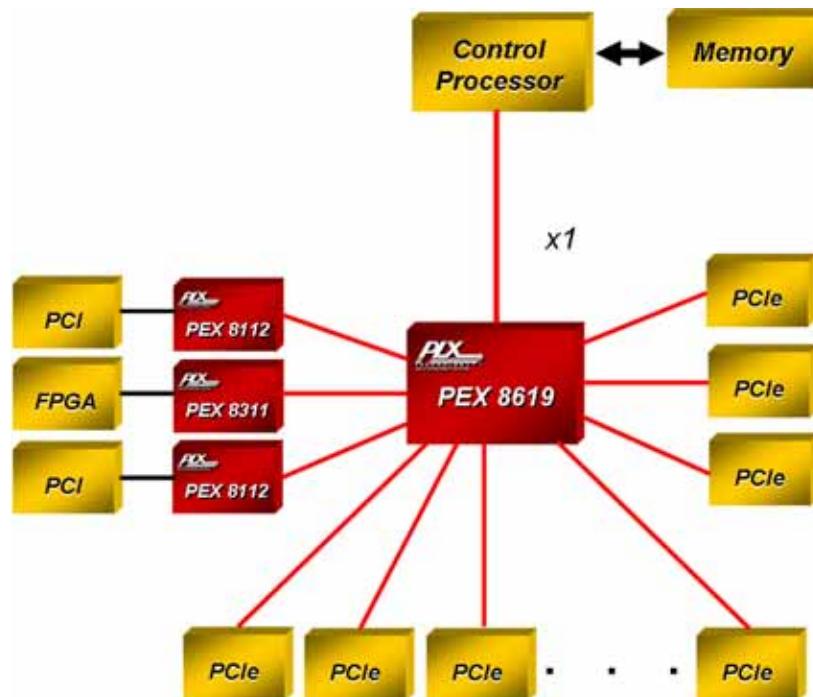
### 2.2.1 Fan-Out

The PEX 8619, with its high Port count and flexible configurations, allows user-specific tuning to a variety of **Host-centric**, as well as **Peer-to-Peer, applications**.

Figure 2-2 illustrates a typical **fan-out** design, where the processor provides a PCI Express Link that must be fanned into a larger quantity of smaller Ports for a variety of I/O functions, each with different bandwidth requirements.

In this example, the PEX 8619 would typically have a 1-Lane upstream Port, and as many as 15 downstream Ports. The downstream Ports provide x1 PCI Express connectivity to the endpoints. With its 16 Ports, the PEX 8619 can provide fan-out connectivity to up to 15 PCI Express devices. Figure 2-2 also illustrates how some of the Ports can be bridged, to provide **PCI slots or generic devices** through the use of the PLX ExpressLane PEX 8311 and PEX 8112 PCI Express bridges.

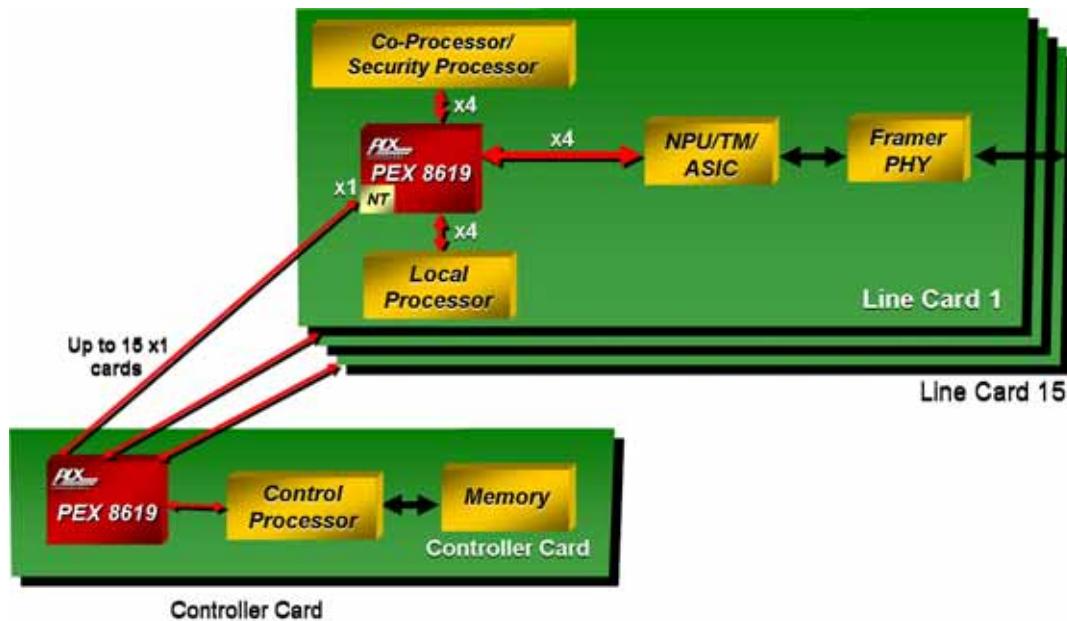
**Figure 2-2. Fan-In/Fan-Out Usage**



## 2.2.2 Control Plane Application

The PEX 8619 is ideal for control planes in routers and other communications sub-systems, to meet increased packet processing needs without compromising latency. Figure 2-3 illustrates a controller card with a PEX 8619 connecting the Control Processor to as many as 15 line cards each, by way of a x1 connection. This usage model provides connectivity to multiple line cards, providing the processor with control over a large quantity of line cards in communication platforms.

**Figure 2-3. Control Plane Application**

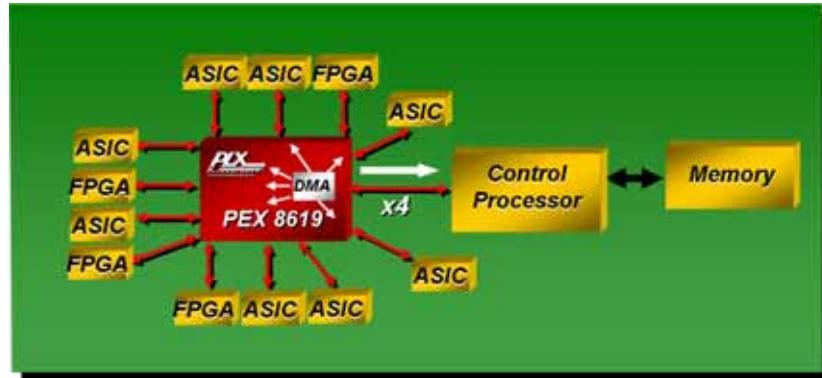


**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 2.2.2.1 DMA in Control Planes for Large Systems – DMA Mode Only

The PEX 8619's DMA engine can alternatively be used to perform housekeeping tasks, *such as* reading device status information from the many endpoints available, as illustrated in Figure 2-4. This particular usage model allows the DMA engine to perform the many Data transfers for endpoint status that would otherwise be handled by the processor.

Figure 2-4. PEX 8619 DMA in a Packet Router



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

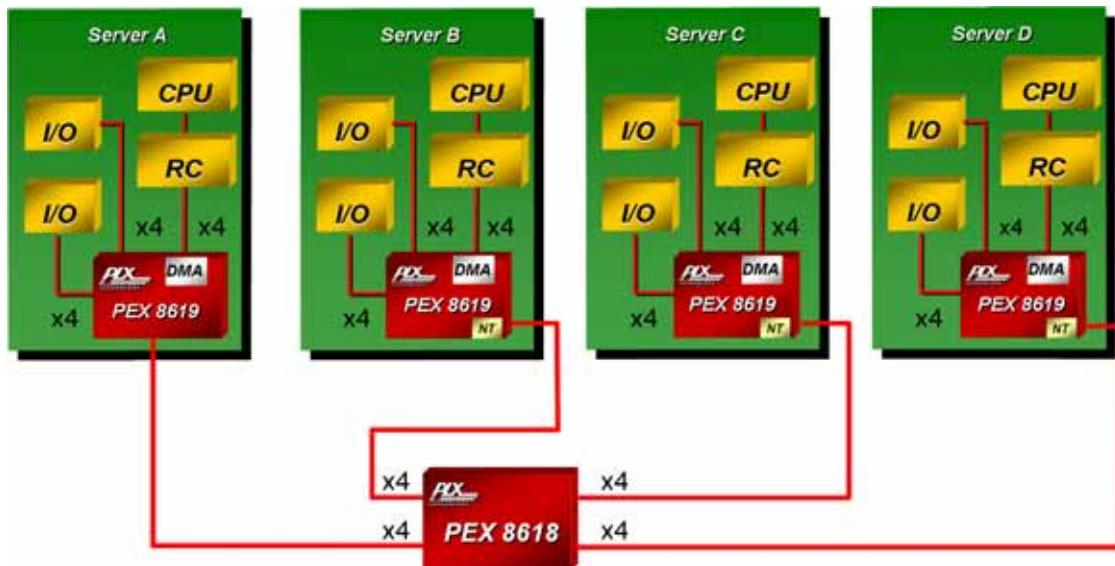
ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 2.2.3 Distributed Computing

The PEX 8619 is well-suited for distributed computing applications, as illustrated in [Figure 2-5](#). Once programmed, the PEX 8619's DMA engine transfers data between the servers and through the Legacy NT Port, at a high rate. The processor can therefore focus all its computing cycles, manipulating the large amounts of data received.

**Figure 2-5. Distributed Computing**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

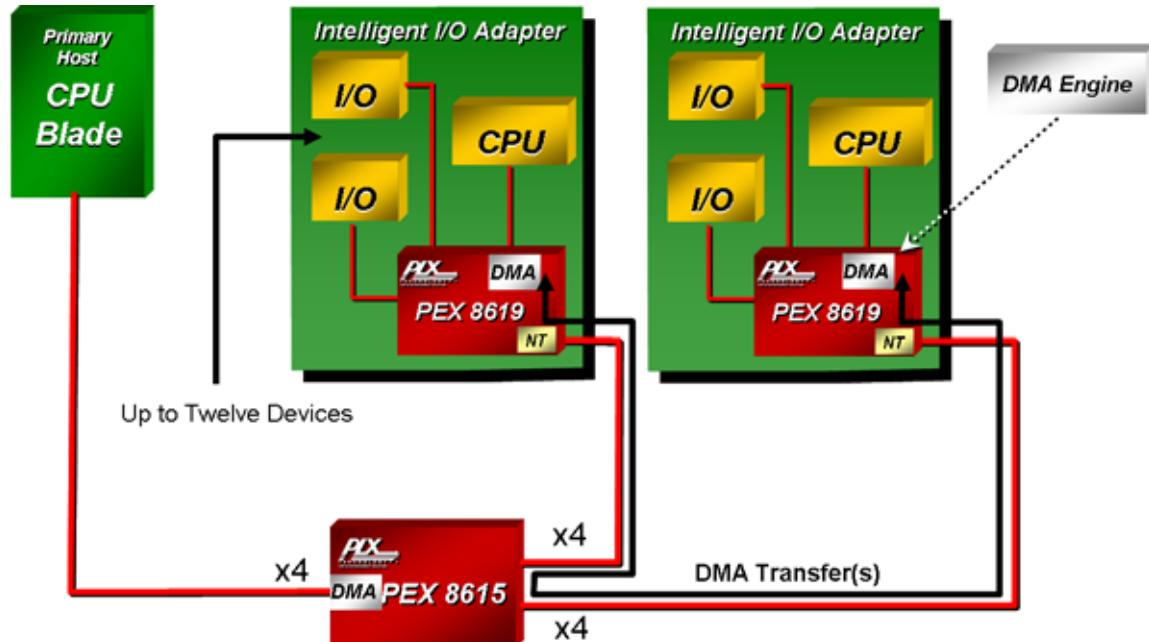
## 2.2.4 Intelligent Adapter Board Redundancy – NT Mode Only

The PEX 8619 supports the **Non-Transparency** (NT) feature. Figure 2-6 illustrates a Host system using an intelligent adapter board, where the adapter board CPU is isolated from the Host CPU.

The PEX 8619 NT Port allows the two CPUs to be isolated, but communicate with one another through registers designed specifically for that purpose. Moreover, the internal DMA engine can be used to copy or mirror the data between adapter boards, so that intelligent I/O adapter redundancy is achieved.

This model can be expanded to an Active-Standby Failover model. In this case, there is an Active Host and a Standby Host, ready to take over the system if the Active Host fails. The PEX 8619's DMA engine can be used to mirror the data between Hosts thus minimizing failover time.

Figure 2-6. Intelligent Adapter Board Usage



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

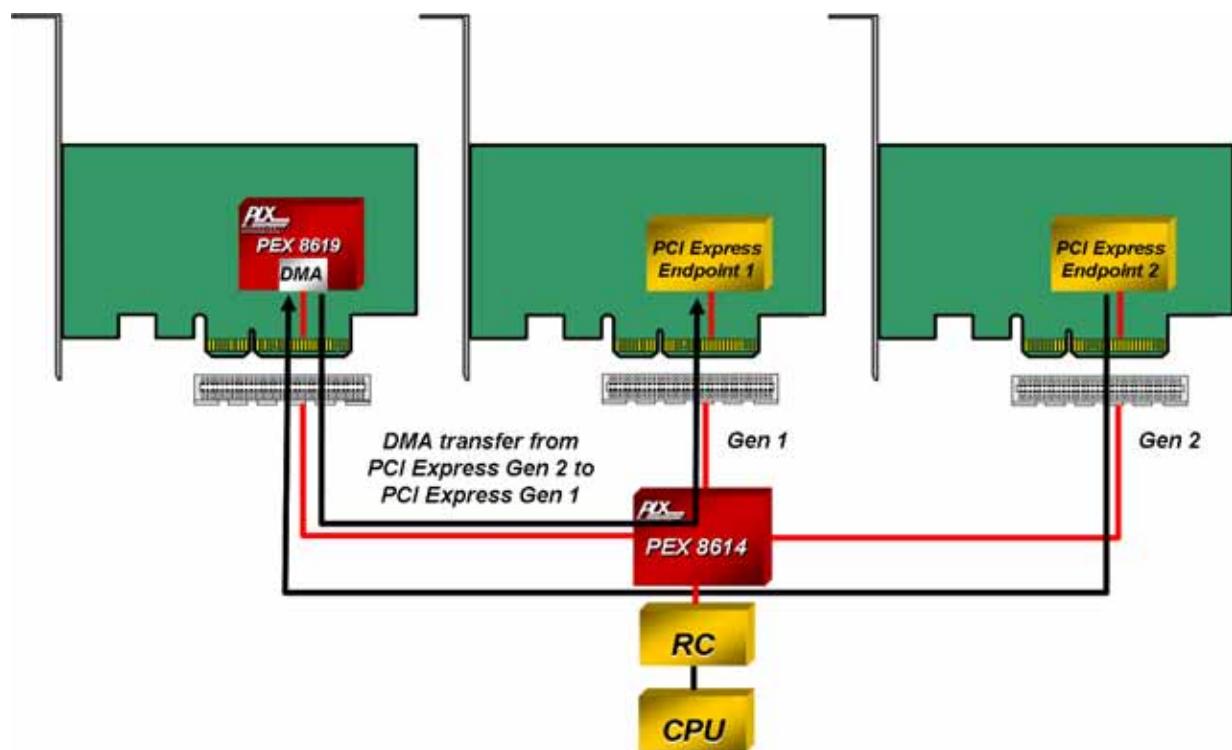
ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 2.2.5 Generic Data Mover

The PEX 8619's flexible DMA engine implements **four DMA channels**, capable of saturating a PCI Express x8 Gen 2 Link. Furthermore, it supports a Descriptor Ring approach, in which the Descriptors can be placed into external Host memory or internal to the PEX 8619. The DMA engine can move data from any Port to any Port, including the same Port (*that is*, the source and destination for a DMA transfer can be the same PEX 8619 Port). [Figure 2-7](#) illustrates two PCI Express endpoints and a PEX 8619, connected through a PCI Express switch. In this example, the PCI Express endpoints take advantage of the PEX 8619's DMA capabilities, for moving data from a PCI Express Gen 2 Link to a PCI Express Gen 1 Link.

**Figure 2-7. Generic Data Mover**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 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 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 and/or Host can configure the other Ports, by way of the upstream Port, using Conventional PCI enumeration.

The PEX 8619's DMA engine is implemented as a separate function in the upstream Port. That is, the PEX 8619 is a multi-function device, where Function 0 (F0) is the Transparent PCI-to-PCI Bridge function and Function 1 (F1) is the DMA Endpoint function. Although there is no special software required for a PCI-to-PCI bridge, the DMA function is an endpoint that requires a driver for programmability and functionality.

### 2.3.1 System Configuration

The virtual PCI-to-PCI bridges within the PEX 8619 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 or Type 1 Configuration Requests, through the virtual primary bus interface (matching Bus Number, Device Number, and Function Number). Assuming the Bus Number for the upstream Port is 1, the PEX 8619's BDF (Bus, Device, Function) for F0 and F1 is 1/0/0 and 1/0/1, respectively.

### 2.3.2 Interrupt Sources and Events

The PEX 8619 supports the INTx Interrupt Message type (compatible with *PCI r3.0* Interrupt signals) or Message Signaled Interrupts (MSIs), when enabled. The PEX 8619 generates interrupts/Messages for the following:

- Hot Plug or Link State events
- PCI Express Hot Plug events
- Device-Specific errors
- GPIO-generated events
- NT Doorbell-generated events
- DMA events (DMA mode only)
- Baseline and Advanced Error Reporting

Internally generated interrupts and interrupts forwarded from downstream Ports are re-mapped and collapsed at the upstream Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 3 Signal Ball Description

### 3.1 Introduction

This chapter provides descriptions of the 324 PEX 8619 signal balls. The signal name, type, location, and a brief description are provided for each signal ball. A map of the PEX 8619's physical layout is also provided.

### 3.2 Abbreviations

The following abbreviations are used in the signal tables provided in this chapter.

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

Abbreviation	Description
#	Active-Low signal
A	Analog Input signal
APWR	Power (VDD10A) balls for SerDes Analog circuits
CMLCLK_CFCn <sup>a</sup>	Spread-Spectrum differential low-voltage, high-speed, CML negative Clock inputs
CMLCLK_CFCp <sup>a</sup>	Spread-Spectrum differential low-voltage, high-speed, CML positive Clock inputs
CMLCLKn <sup>a</sup>	Differential low-voltage, high-speed, CML negative Clock inputs
CMLCLKp <sup>a</sup>	Differential low-voltage, high-speed, CML 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	1.0V Power (VDD10) balls for low-voltage Core circuits
GND	Common Ground (VSS) for all circuits
I	Input
I/O	Bidirectional (Input or Output)
I/OPWR	2.5V Power (VDD25) balls for Input and Output interfaces
O	Output
OD	Open Drain output
PLLPWR	2.5V Power (VDD25A) 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 8619 configuration, operational mode setting, and <i>Factory Test</i> ; these signals generally are not toggled at runtime

a. For REFCLK input, CML source is recommended; however, LVDS source is supported.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 3.3 Internal Pull-Up Resistors

The PEX 8619 contains I/O buffers that have weak internal pull-up resistors, indicated in this chapter by PU 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 [VDD25](#) or Low to Ground ([VSS](#)), as appropriate, through a  $3\text{K}\Omega$  to  $10\text{K}\Omega$  resistor.

[Table 3-2](#) lists the internal pull-up resistor values.

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

Internal Resistor	Minimum	Typical	Maximum	Units
PU	74K	111K	178K	$\Omega$

### 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\_PERn15 is located at M5D7, PEX\_PERn14 is located at L5D8, and so forth.

If there is more than one ball per signal name that does not include a numbered range (such as [VDD10](#)), the locations are listed in ascending alphanumeric order.

The PEX 8619 signals are divided into the following groups:

- [PCI Express Signals](#)
- [Serial Hot Plug Signals](#)
- [Serial EEPROM Signals](#)
- [Strapping Signals](#)
- [JTAG Interface Signals](#)
- [I<sup>2</sup>C/SMBus Slave Interface Signals](#)
- [Device-Specific Signals](#)
- [External Resistor Signals](#)
- [No Connect Signals](#)
- [Power and Ground Signals](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 3.4.1 PCI Express Signals

[Table 3-3](#) defines the PCI Express SerDes and Control signals.

**Table 3-3. PCI Express Signals – 69 Balls**

Signal Name	Type	Location	Description
PEX_PERn[15:0]	CMLRn	M5, L5, J5, H5, D7, D8, D10, D11, G14, H14, K14, L14, R12, R11, R9, R8	<b>Negative Half of PCI Express Receiver Differential Signal Pairs (16 Balls)</b>
PEX_PERp[15:0]	CMLRp	M4, L4, J4, H4, E7, E8, E10, E11, G15, H15, K15, L15, P12, P11, P9, P8	<b>Positive Half of PCI Express Receiver Differential Signal Pairs (16 Balls)</b>
PEX_PERST#	I PU	P17	<b>PCI Express Reset</b> Used to cause a Fundamental Reset. (Refer to <a href="#">Chapter 5, “Reset and Initialization,”</a> for further details.)
PEX_PETn[15:0]	CMLTn	M2, L2, J2, H2, A7, A8, A10, A11, G17, H17, K17, L17, V12, V11, V9, V8	<b>Negative Half of PCI Express Transmitter Differential Signal Pairs (16 Balls)</b>
PEX_PETp[15:0]	CMLTp	M1, L1, J1, H1, B7, B8, B10, B11, G18, H18, K18, L18, U12, U11, U9, U8	<b>Positive Half of PCI Express Transmitter Differential Signal Pairs (16 Balls)</b>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 3-3. PCI Express Signals – 69 Balls (Cont.)**

Signal Name	Type	Location	Description
PEX_REFCLKn	CMLCLKn	V10	<p><b>Negative Half of 100-MHz PCI Express Reference Clock Input Signal Pair</b></p> <p>When Dual Clocking is enabled, the PEX_REFCLKn and PEX_REFCLKp signal balls become the Spread-Spectrum Clocking (SSC) domain Clock signals.</p> <p>PEX_REFCLKn must be AC-coupled.</p> <p>Use a 0.01 to 0.1 <math>\mu</math>F capacitor.</p> <p>Refer to <a href="#">Chapter 19, “Dual Clocking Support,”</a> for further details.</p>
PEX_REFCLKp	CMLCLKp	U10	<p><b>Positive Half of 100-MHz PCI Express Reference Clock Input Signal Pair</b></p> <p>When Dual Clocking is enabled, the PEX_REFCLKn and PEX_REFCLKp signal balls become the SSC domain Clock signals.</p> <p>PEX_REFCLKn must be AC-coupled.</p> <p>Use a 0.01 to 0.1 <math>\mu</math>F capacitor.</p> <p>Refer to <a href="#">Chapter 19, “Dual Clocking Support,”</a> for further details.</p>
PEX_REFCLK_CFCn	CMLCLK_CFCn	A9	<p><b>Negative Half of 100-MHz PCI Express Spread-Spectrum Reference Clock Input Signal Pair</b></p> <p>When Dual Clocking is enabled, the PEX_REFCLK_CFCn and PEX_REFCLK_CFCp signal balls become the Constant Frequency Clocking (CFC) domain Clock signals.</p> <p>Due to internal biasing, the CFC Clock input requires an AC-coupling capacitor. Use a 0.01 to 0.1 <math>\mu</math>F capacitor.</p> <p>Refer to <a href="#">Chapter 19, “Dual Clocking Support,”</a> for further details.</p>
PEX_REFCLK_CFCp	CMLCLK_CFCp	B9	<p><b>Positive Half of 100-MHz PCI Express Spread-Spectrum Reference Clock Input Signal Pair</b></p> <p>When Dual Clocking is enabled, the PEX_REFCLK_CFCn and PEX_REFCLK_CFCp signal balls become the CFC domain Clock signals.</p> <p>Due to internal biasing, the CFC Clock input requires an AC-coupling capacitor. Use a 0.01 to 0.1 <math>\mu</math>F capacitor.</p> <p>Refer to <a href="#">Chapter 19, “Dual Clocking Support,”</a> for further details.</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 3.4.2 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, provided that the PEX 8619 serial EEPROM Sets the **Power Management Hot Plug User Configuration** register *40-Pin I/O Expander Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1E0h[17]**). All Transparent downstream Ports that implement Serial Hot Plug must use the same type of I/O Expander (either 16- or 40-pin, but not both concurrently).

**Table 3-4** defines the three signal balls that support Serial Hot Plug. Additionally, the PEX 8619 supports external Serial Hot Plug signals on the I<sup>2</sup>C I/O Expander. (Refer to [Section 12.6.2, “I<sup>2</sup>C I/O Expander Parts Selection and Pin Definition.”](#))

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

These signals are active only for Hot Plug-capable Ports configured at start-up. For further details regarding Hot Plug, refer to [Chapter 12, “Hot Plug Support.”](#)

**Table 3-4. Serial Hot Plug Signals – 3 Balls**

Signal Name	Type	Location	Description
I2C_SCL1	OD	B16	<p><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 8619 I<sup>2</sup>C Master interface.</p> <p><b>I2C_SCL1 requires an external pull-up resistor.</b></p>
I2C_SDA1	OD	C16	<p><b>I<sup>2</sup>C Serial Data Output 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 8619 I<sup>2</sup>C Master interface.</p> <p><b>I2C_SDA1 requires an external pull-up resistor.</b></p>
SHPC_INT#	I PU	A16	<p><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-enabled 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 corresponding Port’s <b>Slot Status</b> register (offset <b>80h</b>). The I<sup>2</sup>C Master interface halts the reading of I/O Expander registers when the SHPC_INT# input de-asserts.</p> <p><b>SHPC_INT# requires an external pull-up resistor.</b></p> <p><i>Note:</i> SHPC_INT# is internally de-bounced, but must remain stable for at least 10 ms.</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 3.4.3 Serial EEPROM Signals

The PEX 8619 includes four signals for interfacing to a serial EEPROM, defined in [Table 3-5](#). For information regarding serial EEPROM use, refer to [Chapter 6, “Serial EEPROM Controller.”](#)

**Table 3-5. Serial EEPROM Signals – 4 Balls**

Signal Name	Type	Location	Description
EE_CS#	I/O PU	R17	<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	O	N15	<b>PEX 8619 Output to Serial EEPROM Data Input</b>
EE_DO	I/O PU	N16	<b>PEX 8619 Input from Serial EEPROM Data Output</b> Should be pulled High to <b>VDD25</b> . <i>Note:</i> Although this is an I/O signal, its logical operation is input.
EE_SK	I/O PU	R18	<b>Serial EEPROM Clock Frequency Output</b> Programmable, by way of the <b>Serial EEPROM Clock Frequency</b> register <b>EepFreq[2:0]</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>268h[2:0]</b> ), 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 3.4.4 Strapping Signals

The PEX 8619 Strapping signals, defined in [Table 3-6](#), Set the configuration of upstream Port and NT Port assignment, Link width, Spread-Spectrum clocking, and various setup and test modes. These balls must be pulled or tied High to [VDD25](#) or Low to [VSS](#) (GND), as indicated in the table.

After a Fundamental Reset, the [Link Capability](#) (offset [74h](#), in each Port), and [Debug Control](#) and [Port Configuration](#) registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets [1DCh](#) and [574h](#), respectively) capture ball status. Strapping ball Configuration data can be changed by writing new data to these registers from the serial EEPROM. I<sup>2</sup>C can also change Strapping ball Configuration data; however, it should first Set the Port's [Even/Odd Port Disable](#) register *Disable Port x* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [230h\[7:0\]](#), and Odd Ports, offset [234h\[7:0\]](#)), to prevent linkup and Host enumeration. Then, when I<sup>2</sup>C programming is complete, I<sup>2</sup>C should lastly Clear the upstream Port's *Disable Port x* bit, to enable linkup and allow subsequent Host enumeration.

**Table 3-6. Strapping Signals – 31 Balls**

Signal Name	Type	Location	Description
STRAP_DEBUG_SEL0	I	N17	<b>Factory Test Only</b> Must be pulled or tied High to <a href="#">VDD25</a> .
STRAP_FAST_BRINGUP#	I	C2	<b>Factory Test Only</b> Must be pulled or tied High to <a href="#">VDD25</a> .
STRAP_NT_ENABLE#	I	N18	<p><b>Enable NT Mode</b> Active-Low input that enables and disables NT mode.</p> <p>The STRAP_NT_ENABLE# input can be overridden by serial EEPROM and/or I<sup>2</sup>C programming of the <a href="#">Debug Control</a> register <a href="#">NT Mode Enable</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">1DCh[18]</a>). If the register is programmed by serial EEPROM and/or I<sup>2</sup>C, that must be the first serial EEPROM entry, or the first register written by I<sup>2</sup>C, with one exception. (Refer to the Note.)</p> <p>I<sup>2</sup>C can select a Port to be the upstream NT Port, by writing to the <a href="#">Debug Control</a> register, after the upstream Port Link is brought down, by Setting the upstream device's <a href="#">Link Control</a> register <i>Link Disable</i> bit (PCI Express Capability, offset <a href="#">10h[4]</a>). After I<sup>2</sup>C configures the PEX 8619, the upstream Link can be restored by Clearing the <i>Link Disable</i> bit in the device connected to the upstream Port.</p> <p>Software can enable or disable NT mode, by writing to the <a href="#">Debug Control</a> register, if the register's <a href="#">Hardware/Software Configuration Mode Control</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">1DCh[15]</a>) is already Set. The mode will change, following subsequent Hot Reset (or DL_Down condition) at the PEX 8619's upstream Port.</p> <p>L = Enables NT mode H = Disables NT mode (default, if input is not connected)</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 3-6. Strapping Signals – 31 Balls (Cont.)**

Signal Name	Type	Location	Description
STRAP_NT_P2P_EN#	I PU	R5	<p><b>NT PCI-to-PCI Bridge Enable</b></p> <p><b>Note:</b> If NT mode is enabled (<a href="#">STRAP_NT_ENABLE#=L</a>), this input should be pulled or tied Low to <a href="#">VSS</a> (GND), unless the NT PCI-to-PCI bridge between the internal Virtual PCI Bus and the NT Port Virtual Interface must be disabled for software compatibility to earlier NT mode switches.</p> <p>If NT mode is not enabled, optionally, this input can remain unconnected, because the internal pull-up resistor holds the input High.</p> <p>Allows the NT function to be logically placed on the internal Virtual PCI Bus, or behind the PCI-to-PCI bridge for that Port.</p> <p>This input maps to the <a href="#">Debug Control</a> register <a href="#">NT P2P Enable</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">1DCh[6]</a>). This signal and its corresponding register bit must <b>not</b> be toggled at runtime.</p> <p><b>It is important that STRAP_NT_P2P_EN# is pulled or tied High or Low with a resistor. Active drivers should not be used, because STRAP_NT_P2P_EN# is used as an output in certain modes of operation.</b></p> <p>L = Enables NT PCI-to-PCI bridge mode, if NT mode is enabled (pulled or tied Low to <a href="#">VSS</a> (GND))  H = Disables NT PCI-to-PCI bridge mode, if NT mode is enabled (default; the PEX 8619 is in Legacy NT mode, if NT mode is enabled) (pulled or tied High to <a href="#">VDD25</a>)</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 3-6. Strapping Signals – 31 Balls (Cont.)**

Signal Name	Type	Location	Description
STRAP_NT_UPSTRM_PORTSEL[3:0]	I	P6, R1, P2, P1	<p><b>Select Upstream Non-Transparent Port (4 Balls)</b></p> <p>Select any Port as the upstream NT Port.</p> <p>The STRAP_NT_UPSTRM_PORTSEL[3:0] inputs can be overridden by the serial EEPROM value for the <b>Debug Control</b> register <i>NT Port Number</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[27:24]). If the <b>Debug Control</b> register is programmed by serial EEPROM, that must be the first serial EEPROM entry.</p> <p>I<sup>2</sup>C can also change which Port is configured to be the NT Port, by writing to the <b>Debug Control</b> register. I<sup>2</sup>C should first Set the Port's <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), to prevent linkup and Host enumeration. After I<sup>2</sup>C re-configures the PEX 8619, I<sup>2</sup>C should lastly Clear the upstream Port's <i>Disable Port x</i> bit, to enable linkup and allow subsequent Host enumeration.</p> <p>Software can also change which Port is configured to be the NT Port, by writing to the <b>Debug Control</b> register, if the register's <b>Hardware/Software Configuration Mode Control</b> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[15]) is already Set.</p> <p>Refer to <a href="#">Section 15.10, “Port Programmability,”</a> for further details.</p> <p>LLLL = Port 0      LLLH = Port 1      LLHL = Port 2      LLHH = Port 3      LHLL = Port 4      LHLH = Port 5      LHHL = Port 6      LHHH = Port 7      HLLL = Port 8      HLLH = Port 9      HLHL = Port 10      HLHH = Port 11      HHLL = Port 12      HHLH = Port 13      HHHL = Port 14      HHHH = Port 15</p> <p><b>Note:</b> If NT mode is not used (STRAP_NT_ENABLE#=H) and/or the serial EEPROM and/or I<sup>2</sup>C programs NT mode (<b>Debug Control</b> register <i>NT Mode Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[18]) is Set (overriding STRAP_NT_ENABLE#)), value should be HH and/or if the <b>Debug Control</b> register is programmed by serial EEPROM (overriding STRAP_NT_UPSTRM_PORTSEL[3:0]), field [27:24] should be programmed to Fh.</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 3-6. Strapping Signals – 31 Balls (Cont.)**

Signal Name	Type	Location	Description						
STRAP_PLL_BYPASS#	I	R15	<p><b>Factory Test Only</b> Must be pulled or tied High to <b>VDD25</b>.</p>						
STRAP_PORTCFG[3:0]	I	C13, B2, A4, A5	<p><b>Strapping Signals to Select Port Configuration (Quantity of Enabled Ports (2, 3, 4, 6, 7, 9, 10, 13, or 16), and Maximum Quantity of Lanes for Each Specific Port) (4 Balls)</b></p> <p>Defines the enabled Port Numbers and their Link widths. Programs the <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>574h[3:0]</b>) default value.</p> <p>LLLL = x1, x1      LLLH = x4, x1, x1      LLHL = x4, x4, x1, x1, x1, x1, x1, x1, x1, x1, x1      LLHH = x4, x4, x4, x1, x1, x1, x1      LHLL = x4, x4, x4, x4      LHLH = x8, x1, x1, x1, x1, x1, x1, x1, x1      LHHL = x8, x4, x1, x1, x1, x1      LHHH = x8, x4, x4      HLLL = x8, x8      All other encodings are <i>reserved</i>.</p>						
STRAP_PROBE_MODE#	I	T17	<p><b>Factory Test Only</b> Must be pulled or tied High to <b>VDD25</b>.</p>						
STRAP_RESERVED17#	I	F1	<p><b>Factory Test Only</b> Must be pulled or tied High to <b>VDD25</b>.</p>						
STRAP_RESERVED16	I	D14	<p><b>Factory Test Only</b> Must be tied directly to <b>VSS</b> (GND).</p>						
STRAP_SERDES_MODE_EN#	I	C4	<p><b>Factory Test Only</b> Must be pulled or tied High to <b>VDD25</b>.</p>						
STRAP_SMBUS_EN#	I PU	U4	<p><b>System Management Bus Enable</b> Selects the I<sup>2</sup>C or SMBus protocol, and defines the default <b>SMBus Configuration</b> register <b>SMBus Enable</b> bit (Port 0, offset <b>344h[0]</b>, respectively) value, as listed below.</p> <table border="1"> <thead> <tr> <th>STRAP_SMBUS_EN# Input State</th><th>SMBus Enable Value</th></tr> </thead> <tbody> <tr> <td>0</td><td>1</td></tr> <tr> <td>1</td><td>0</td></tr> </tbody> </table> <p>When pulled or tied Low to Ground, enables SMBus Slave protocol on the <b>I2C_SCL0</b> and <b>I2C_SDA0</b> 2-wire bus. ARP is enabled if I2C_ADDR2 is Low when sampled at the time <b>PEX_PERST#</b> input de-asserts.</p> <p>When pulled or tied High to <b>VDD25</b>, enables I<sup>2</sup>C Slave protocol on the I2C_SCL0 and I2C_SDA0 2-wire bus.</p>	STRAP_SMBUS_EN# Input State	SMBus Enable Value	0	1	1	0
STRAP_SMBUS_EN# Input State	SMBus Enable Value								
0	1								
1	0								
STRAP_SPARE[5, 1]#	I/O PU	T5, T4	<p><b>Factory Test Only (2 Balls)</b> Pull or tie High to <b>VDD25</b> for standard operation.</p>						

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 3-6. Strapping Signals – 31 Balls (Cont.)**

Signal Name	Type	Location	Description
STRAP_SPARE0#	O PU	R4	<b>Factory Test Only</b> Do not connect this ball to board electrical paths.
STRAP_SSC_ISO_ENABLE#	I	U3	<b>Spread Spectrum Clocking (SSC) Crossing Enable</b> Allows the SSC REFCLK clock source on upstream Port 0, with a x8 Link width and Constant Frequency Clocking (CFC) REFCLK source on downstream devices. (Refer to Chapter 19, “Dual Clocking Support,” for further details.) L = Enabled (pulled or tied Low to <b>VSS</b> (GND)) H = Disabled (pulled or tied High to <b>VDD25</b> )
STRAP_TESTMODE[3:0]	I	U17, V16, V15, V14	<b>Factory Test Only (4 Balls)</b> Must be pulled or tied High to <b>VDD25</b> .
STRAP_UPCFG_TIMER_EN#	I PU	U5	<b>Link Upconfigure Timer Enable</b> This input maps to the <b>Debug Control</b> register <i>UPCFG Timer Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1DCh[4]</b> ). STRAP_UPCFG_TIMER_EN# and the <i>UPCFG Timer Enable</i> bit must <b>not</b> be toggled at runtime. When STRAP_UPCFG_TIMER_EN# is pulled or tied High to <b>VDD25</b> , the Data Rate Identifier symbol in the TS Ordered-Sets always advertises support for both the Gen 2 data rate and Autonomous Change. When STRAP_UPCFG_TIMER_EN# is pulled or tied Low to <b>VSS</b> (GND), if this Link training sequence fails during the Configuration 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. If Link training continues to fail when the LTSSM is in the <i>Configuration</i> state, the LTSSM continues to alternate between Gen 1 and Gen 2 advertisement every time it exits the <i>Detect</i> state. <i>Note: This feature should be enabled only if a non-compliant device will not linkup if these Data Rate Identifier bits are Set.</i>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 3-6. Strapping Signals – 31 Balls (Cont.)**

Signal Name	Type	Location	Description
STRAP_UPSTRM_PORTSEL[3:0]	I	E1, F2, F3, E2	<p><b>Strapping Signals to Select Upstream Port (4 Balls)</b>  The STRAP_UPSTRM_PORTSEL[3:0] inputs can be overridden by the serial EEPROM value for the <b>Debug Control</b> register <i>Upstream Port ID</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[11:8]). <b>If the Debug Control register is programmed by serial EEPROM, that must be the first serial EEPROM entry.</b></p> <p><math>I^2C</math> can also change which Port is configured to be the upstream Port, by writing to the <b>Debug Control</b> register. <math>I^2C</math> should first Set the Port's <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), to prevent linkup and Host enumeration. After <math>I^2C</math> re-configures the PEX 8619, <math>I^2C</math> should lastly Clear the upstream Port's <i>Disable Port x</i> bit, to enable linkup and allow subsequent Host enumeration.</p> <p>Software can also change which Port is configured to be the upstream Port, by writing to the <b>Debug Control</b> register, if the register's <i>Hardware/Software Configuration Mode Control</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[15]) is already Set.</p> <p>Refer to <a href="#">Section 15.10, “Port Programmability,”</a> for further details.</p> <p>LLLL = Port 0  LLLH = Port 1  LLHL = Port 2  LLHH = Port 3  LHLL = Port 4  LHLH = Port 5  LHHL = Port 6  LHHH = Port 7  HLLL = Port 8  HLLH = Port 9  HLHL = Port 10  HLHH = Port 11  HHLL = Port 12  HHLH = Port 13  HHHL = Port 14  HHHH = Port 15</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 3.4.5 JTAG Interface Signals

The PEX 8619 includes five signals for performing Joint Test Action Group (JTAG) boundary scan, defined in [Table 3-7](#). The JTAG interface is described in [Section 20.8, “JTAG Interface.”](#)

**Table 3-7. JTAG Interface Signals – 5 Balls**

Signal Name	Type	Location	Description
JTAG_TCK	I	B15	<b>JTAG Test Clock Input</b> JTAG Test Access Port (TAP) Controller clock source. Frequency can be from 0 to 20 MHz.
JTAG_TDI	I	E14	<b>JTAG Test Data Input</b> Serial input to the JTAG TAP Controller, for test instructions and data.
JTAG_TDO	O	C15	<b>JTAG Test Data Output</b> Serial output from the JTAG TAP Controller test instructions and data.
JTAG_TMS	I	C14	<b>JTAG Test Mode Select</b> Input decoded by the JTAG TAP Controller, to control test operations.
JTAG_TRST#	I	D15	<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 <b>VSS</b> (GND) through a $1.5\text{K}\Omega$ 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

Table 3-8 defines the five signals that support the I<sup>2</sup>C/SMBus Slave interface. For further details, refer to Chapter 7, “I<sup>2</sup>C/SMBus Slave Interface Operation.”

**Table 3-8. I<sup>2</sup>C/SMBus Slave Interface Signals – 5 Balls**

Signal Name	Type	Location	Description																													
I2C_ADDR[2:0]	I	E18, E17, C18	<p><b>I<sup>2</sup>C/SMBus Slave Address Bits 2 through 0 Inputs (3 Pins)</b></p> <p>Used to define the default value of the three least significant bits of the PEX 8619 I<sup>2</sup>C/SMBus 7-bit Slave address, which is programmable in the <b>I<sup>2</sup>C Configuration</b> register <b>Slave Address</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 294h[2:0], default value is 38h). If I<sup>2</sup>C or SMBus configuration is used, I2C_ADDR[2:0] 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) that have the upper four bits of their 7-bit I<sup>2</sup>C/SMBus Slave address also defined as value 0111b.</p> <p>If I2C_ADDR[2:0] are left unconnected, the value of the lowest three bits is 111b (due to the internal pull-up resistors), and the 7-bit Slave address defaults to value 0111_111b. The eight possible default values for the Slave address are listed below.</p> <table border="1"> <thead> <tr> <th>I2C_ADDR[2:0] Values</th><th>Offset 294h[2:0] Value</th><th>Offset 294h[6:3] Value</th><th>I<sup>2</sup>C/SMBus Slave Address</th></tr> </thead> <tbody> <tr> <td>000</td><td>000b</td><td rowspan="8">0111b</td><td>0111_000b</td></tr> <tr> <td>001</td><td>001b</td><td>0111_001b</td></tr> <tr> <td>010</td><td>010b</td><td>0111_010b</td></tr> <tr> <td>011</td><td>011b</td><td>0111_011b</td></tr> <tr> <td>100</td><td>100b</td><td>0111_100b</td></tr> <tr> <td>101</td><td>101b</td><td>0111_101b</td></tr> <tr> <td>110</td><td>110b</td><td>0111_110b</td></tr> <tr> <td>111</td><td>111b</td><td>0111_111b</td></tr> </tbody> </table> <p>When SMBus is enabled (<b>SMBus Configuration</b> register <b>SMBus Enable</b> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[0]) is Set (the default value when <b>STRAP_SMBUS_EN#</b> is sampled and latched Low at <b>PEX_PERST#</b> input de-assertion):</p> <ul style="list-style-type: none"> <li>• I2C_ADDR[1:0] define the two least significant bits of the SMBus Slave Address,</li> <li>• Bit 2 of the SMBus Slave Address defaults to value 0, and</li> <li>• I2C_ADDR0 input defines the default value of the <b>SMBus Configuration</b> register <b>ARP Disable</b> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[8]), to enable (0) or disable (1) SMBus Address Resolution Protocol (ARP).</li> </ul> <p><b>I2C_ADDR[2:0] require external pull-up or pull-down termination resistors. If I<sup>2</sup>C is not used, external termination is strongly recommended, to prevent possible Input buffer oscillation.</b></p>	I2C_ADDR[2:0] Values	Offset 294h[2:0] Value	Offset 294h[6:3] Value	I <sup>2</sup> C/SMBus Slave Address	000	000b	0111b	0111_000b	001	001b	0111_001b	010	010b	0111_010b	011	011b	0111_011b	100	100b	0111_100b	101	101b	0111_101b	110	110b	0111_110b	111	111b	0111_111b
I2C_ADDR[2:0] Values	Offset 294h[2:0] Value	Offset 294h[6:3] Value	I <sup>2</sup> C/SMBus Slave Address																													
000	000b	0111b	0111_000b																													
001	001b		0111_001b																													
010	010b		0111_010b																													
011	011b		0111_011b																													
100	100b		0111_100b																													
101	101b		0111_101b																													
110	110b		0111_110b																													
111	111b		0111_111b																													

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 3-8. I<sup>2</sup>C/SMBus Slave Interface Signals – 5 Balls (Cont.)**

Signal Name	Type	Location	Description
I2C_SCL0	OD	C17	<p><b>I<sup>2</sup>C/SMBus Serial Clock Line</b>  I<sup>2</sup>C/SMBus Clock line. Data on the I<sup>2</sup>C Bus can be transferred at rates of up to 100 kbit/s (Standard mode).</p> <p><b>I2C_SCL0 requires an external pull-up resistor.</b></p> <p><i>Note:</i> The PEX 8619 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 8619 registers is being processed.</p>
I2C_SDA0	OD	A17	<p><b>I<sup>2</sup>C/SMBus Serial Data I/O</b>  Transmits and receives I<sup>2</sup>C/SMBus data during I<sup>2</sup>C/SMBus accesses to PEX 8619 registers.</p> <p><b>I2C_SDA0 requires an external pull-up resistor.</b></p>

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 3.4.7 Device-Specific Signals

Table 3-9 defines the Device-Specific signals – signals that are unique to the PEX 8619.

**Table 3-9. Device-Specific Signals – 51 Balls**

Signal Name	Type	Location	Description
FATAL_ERR#	O	B13	<p><b>Fatal Error Output</b>            Asserted Low when a Fatal error is detected in the PEX 8619 and the following conditions exist (all the same conditions that are required to send a Fatal Error Message to the Host):</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>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>
GPIO[31:0]	I/O PU	D4, D5, B4, B5, C5, C3, B14, E15, D16, T18, P15, R16, R14, T15, T16, P14, V6, V5, V4, R6, U6, U2, P5, P4, R3, T3, F4, D1, D2, E3, D3, E4	<p><b>General-Purpose I/O (32 Balls)</b>            Default functionality is determined at Fundamental Reset; however, functionality can be switched by programming the <b>GPIO</b> registers by way of serial EEPROM, I<sup>2</sup>C/SMBus, and/or software. GPIO[31:0] provide GPIO input functionality, by default, when the <b>STRAP_TESTMODE[3:0]</b> inputs, sampled at Fundamental Reset (<b>PEX_PERST#</b> input de-assertion), are a value of 1111b (Fh). Alternatively, when the <b>STRAP_TESTMODE[3:0]</b> inputs, sampled at Fundamental Reset, are a value of 1100b (Ch), GPIO[15:0] function as Serial Hot Plug <b>PERST#</b> Reset outputs, by default, for the corresponding Ports that include an external I<sup>2</sup>C I/O Expander. If an external I<sup>2</sup>C I/O Expander is not present for a Port, the corresponding GPIO[15:0] output remains Low (the Serial Hot Plug PERST# output for that Port is <i>not</i> de-asserted). If Serial Hot Plug is implemented (using external I<sup>2</sup>C I/O Expanders), it is recommended that the GPIO[15:0] signals be strapped as Serial Hot Plug PERST# Reset outputs and routed to the slots, rather than using the PERST# outputs from the I<sup>2</sup>C I/O Expanders.</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 3-9. Device-Specific Signals – 51 Balls (Cont.)**

Signal Name	Type	Location	Description
PEX_INTA#	OD	P18	<p><b>Interrupt Output</b></p> <p>PEX_INTA# Interrupt output is enabled if:</p> <ul style="list-style-type: none"> <li>INTx Messages are enabled (<b>PCI Command</b> register <i>Interrupt Disable</i> bit, offset <b>04h[10]</b>, is Cleared), and MSIs are disabled (<b>MSI Control</b> register <i>MSI Enable</i> bit, offset <b>48h[16]</b>, is Cleared)</li> <li>PEX_INTA# output (<b>ECC Error Check Disable</b> register <i>Enable PEX_INTA# Interrupt Output(s) for x Interrupt</i> bit(s), offset <b>1C8h[7, 6, 5, and/or 4]</b>) is enabled (refer to the register description, for Port associations)</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 errors</li> <li>Device-Specific NT Port Link Interface errors and events</li> <li>NT-Virtual Doorbell events</li> <li>NT-Link Doorbell events</li> </ul> <p>Refer to <a href="#">Section 11.4, “PEX_INTA# Interrupts,”</a> for details.</p>
PEX_LANE_GOOD[15:0]#	I/O PU	T2, T1, P3, R2, E5, E13, A13, B3, E16, D17, D18, B17, U15, U16, T14, U14	<p><b>Active-Low PCI Express Lane Status Indicator Outputs for Lanes [15:0] (16 Balls)</b></p> <p>Directly drives the common-anode LED module. PEX_LANE_GOOD[15:0]# LED behavior:</p> <ul style="list-style-type: none"> <li><b>Off</b> – Lane is disabled</li> <li><b>On</b> – Lane is enabled, 5.0 GT/s</li> <li><b>Blinking, 0.5 seconds On, 0.5 seconds Off</b> – Lane is enabled, 2.5 GT/s, reduced Lanes are up</li> </ul>
PEX_NT_RESET#	O	P16	<p><b>Active-Low Output Used to Propagate Reset in NT Mode</b></p> <p>Pulse width is 1 <math>\mu</math>s.</p>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 3.4.8 External Resistor Signals

**Table 3-10. External Resistor Signals – 8 Balls**

Signal Name	Type	Location	Description
REXT_A[3:0]	A	K4, D9, J15, R10	<b>External Resistor Balls (4 Balls)</b> One pair per SerDes block (paired with the “B” signal). <b>An external 1.43KΩ 1% resistor must be attached between each REXT_A and REXT_B pair.</b> <b>Do not connect to any other signal, power, nor ground.</b>
REXT_B[3:0]	A	L6, F8, H13, N11	<b>External Resistor Balls (4 Balls)</b> One pair per SerDes block (paired with the “A” signal). <b>An external 1.43KΩ 1% resistor must be attached between each REXT_A and REXT_B pair.</b> <b>Do not connect to any other signal, power, nor ground.</b>

### 3.4.9 No Connect Signals

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

**Table 3-11. No Connect Signals – 11 Balls**

Signal Name	Type	Location	Description
N/C	<b>Reserved</b>	E9, J14, J17, J18, K1, K2, K5, P10, T6	<b>No Connect (9 Balls)</b> Do not connect these balls to board electrical paths.
THERMAL_DIODEn	<b>Reserved</b>	A15	<b>No Connect</b> <b>Factory Test Only</b> Do not connect this ball to board electrical paths.
THERMAL_DIODEp	<b>Reserved</b>	A14	<b>No Connect</b> <b>Factory Test Only</b> Do not connect this ball to board electrical paths.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 3.4.10 Power and Ground Signals

**Table 3-12. Power and Ground Signals – 137 Balls**

Signal Name	Type	Location	Description
VDD10	CPWR	F7, F11, F12, G6, G13, H6, L13, M6, M13, N7, N8, N12	<b>1.0V ±5% Power for Core and SerDes Digital Logic (12 Balls)</b>
VDD10A	APWR	C7, C9, C11, G16, H3, J16, K3, L16, M3, T8, T10, T12	<b>1.0V ±5% Power for SerDes Analog Circuits (12 Balls)</b>
VDD25	I/OPWR	F6, F13, N6, N13	<b>2.5V ±10% Power for I/O Logic Functions (4 Balls)</b>
VDD25A	PLLPWR	F9, J13, K6, N10	<b>2.5V ±10% Power for Phase-Locked Loop (PLL) Circuits (4 Balls)</b>
VSS	GND	A1, A2, A3, A6, A12, A18, B1, B6, B12, B18, C1, C6, C8, C10, C12, D6, D12, D13, E6, E12, F5, F10, F14, F15, F16, F17, F18, G1, G2, G3, G4, G5, H16, J3, J6, K13, K16, L3, M14, M15, M16, M17, M18, N1, N2, N3, N4, N5, N9, N14, P7, P13, R7, R13, T7, T9, T11, T13, U1, U7, U13, U18, V1, V2, V3, V7, V13, V17, V18	<b>Ground Connections (69 Balls)</b>
VSS_THERMAL	Thermal-GND	G7, G8, G9, G10, G11, G12, H7, H8, H9, H10, H11, H12, J7, J8, J9, J10, J11, J12, K7, K8, K9, K10, K11, K12, L7, L8, L9, L10, L11, L12, M7, M8, M9, M10, M11, M12	<b>Thermal-Ball Ground Connection (36 Balls)</b>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 3.5 Physical Layout

**Figure 3-1. PEX 8619 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
A	VSS	VSS	VSS	STRAP_P ORTCFG1	STRAP_P ORTCFG0	VSS	PEX_PET n11	PEX_PET n10	PEX_REF CLK_CFC n	PEX_PET n9	PEX_PET n8	VSS	PEX_LAN E_GOOD9 #	THERMAL _DIODEp	THERMAL _DIODEn	SHPC_INT #	I2C_SDA0	VSS
B	VSS	STRAP_P ORTCFG2	PEX_LAN E_GOOD8 #	GPIO29	GPIO28	VSS	PEX_PET p11	PEX_PET p10	PEX_REF CLK_CFC p	PEX_PET p9	PEX_PET p8	VSS	FATAL_E RR#	GPIO25	JTAG_TC K	I2C_SCL1	PEX_LAN E_GOOD4 #	VSS
C	VSS	STRAP_F AST_BRN GUP#	GPIO26	STRAP_S ERDES_M ODE_EN#	GPIO27	VSS	VDD10A	VSS	VDD10A	VSS	VDD10A	VSS	STRAP_P ORTCFG3	JTAG_TM S	JTAG_TD O	I2C_SDA1	I2C_SCL0	I2C_ADDR 0
D	GPIO4	GPIO3	GPIO1	GPIO31	GPIO30	VSS	PEX_PER n11	PEX_PER n10	REXT_A2	PEX_PER n9	PEX_PER n8	VSS	VSS	STRAP_R ESERVED 16	JTAG_TR ST#	GPIO23	PEX_LAN E_GOOD6 #	PEX_LAN E_GOOD5 #
E	STRAP_P STRM_PO RTSEL3	STRAP_U PSTRM_P RTSEL0	GPIO2	GPIO0	PEX_LAN E_GOOD1 1#	VSS	PEX_PER p11	PEX_PER p10	N/C	PEX_PER p9	PEX_PER p8	VSS	PEX_LAN E_GOOD1 0#	JTAG_TDI	GPIO24	PEX_LAN E_GOOD7 #	I2C_ADDR 1	I2C_ADDR 2
F	STRAP_R ESERVED 17#	STRAP_P STRM_PO RTSEL2	GPIO5	VSS	VDD25	VDD10	REXT_B2	VDD25A	VSS	VDD10	VDD10	VDD25	VSS	VSS	VSS	VSS	VSS	
G	VSS	VSS	VSS	VSS	VDD10	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VDD10	PEX_PER n7	PEX_PER p7	VDD10A	PEX_PET n7	PEX_PET p7	
H	PEX_PET p12	PEX_PET n12	VDD10A	PEX_PER p12	PEX_PER n12	VDD10	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	REXT_B1	PEX_PER n6	PEX_PER p6	VSS	PEX_PET n6	PEX_PET p6	
J	PEX_PET p13	PEX_PET n13	VSS	PEX_PER p13	PEX_PER n13	VSS	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VDD25A	N/C	REXT_A1	VDD10A	N/C	N/C	
K	N/C	N/C	VDD10A	REXT_A3	N/C	VDD25A	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VDD10	PEX_PER n5	PEX_PER p5	VSS	PEX_PET n5	PEX_PET p5	
L	PEX_PET p14	PEX_PET n14	VSS	PEX_PER p14	PEX_PER n14	REXT_B3	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VDD10	PEX_PER n4	PEX_PER p4	VDD10A	PEX_PET n4	PEX_PET p4	
M	PEX_PET p15	PEX_PET n15	VDD10A	PEX_PER p15	PEX_PER n15	VDD10	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VSS_THE RMAL	VDD10	VSS	VSS	VSS	VSS	VSS	
N	VSS	VSS	VSS	VSS	VSS	VDD25	VDD10	VDD10	VSS	VDD25A	REXT_B0	VDD10	VDD25	VSS	EE_DI	EE_DO	STRAP_D EBUG_SE L0	STRAP_N T_ENABL E#
P	STRAP_N T_UPSTR M_PORTS ELO	STRAP_N T_UPSTR M_PORTS E1	PEX_LAN E_GOOD1 3#	GPIO8	GPIO9	STRAP_N T_UPSTR M_PORTS EL3	VSS	PEX_PER p0	PEX_PER p1	N/C	PEX_PER p2	PEX_PER p3	VSS	GPIO16	GPIO21	PEX_NT RESET#	PEX_PER ST#	PEX_INTA #
R	STRAP_N T_UPSTR M_PORTS ELO	PEX_LAN E_GOOD1 2#	GPIO7	STRAP_S PARE0#	STRAP_N T_P2P_EN #	GPIO12	VSS	PEX_PER n0	PEX_PER n1	REXT_A0	PEX_PER n2	PEX_PER n3	VSS	GPIO19	STRAP_P LL_BYPAS S#	GPIO20	EE_CS#	EE_SK
T	PEX_LAN E_GOOD1 4#	PEX_LAN E_GOOD1 5#	GPIO6	STRAP_S PARE1#	STRAP_S PARE5#	N/C	VSS	VDD10A	VSS	VDD10A	VSS	VDD10A	PEX_LAN E_GOOD1	GPIO18	GPIO17	STRAP_P ROBE_M ODE#	GPIO22	
U	VSS	GPIO10	STRAP_S SC_ISO_E NABLE#	STRAP_S MBUS_EN #	STRAP_U PCFG_TI MER_EN#	GPIO11	VSS	PEX_PET p0	PEX_PET p1	PEX_REF CLKp	PEX_PET p2	PEX_PET p3	VSS	PEX_LAN E_GOOD0 #	PEX_LAN E_GOOD3 #	PEX_LAN E_GOOD2 #	STRAP_T ESTMODE 3	
V	VSS	VSS	VSS	GPIO13	GPIO14	GPIO15	VSS	PEX_PET n0	PEX_PET n1	PEX_REF CLKn	PEX_PET n2	PEX_PET n3	VSS	STRAP_T ESTMODE 0	STRAP_T ESTMODE 1	STRAP_T ESTMODE 2	VSS	VSS

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

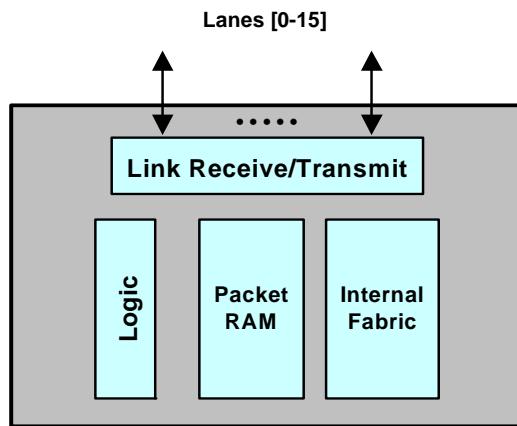
ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 4.1 Hardware Architecture

The PEX 8619 is designed with a flexible, modular architecture. The 16 PCI Express Lanes are connected to one another by the internal fabric to the central RAM. [Figure 4-1](#) provides a block diagram of the PEX 8619.

**Figure 4-1. PEX 8619 Block Diagram**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 4.1.1 Port Functions

Each Port implements the *PCI Express Base r2.0* Physical, Data Link, and Transaction Layers (PHY, DLL, and TL, respectively). The PEX 8619 supports 16 integrated Serializer/De-Serializer (SerDes) modules, which provide the 16 PCI Express hardware interface Lanes. The Lanes can be combined, for a maximum of 16 Ports.

### 4.1.1.1 Port Configurations

The upstream and downstream Ports' Link widths are initially defined by the **STRAP\_PORTCFG[3:0]** inputs, which are pulled or tied High to **VDD25** or Low to **VSS** (GND). 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\_PORTCFG[3:0]** inputs at that time. Port configuration can also be changed through the I<sup>2</sup>C Slave interface. 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. A x8 Port can negotiate down to 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 x8 Port that cannot train to x8 attempts to negotiate down to 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 (2.5 GT/s) or Gen 2 (5.0 GT/s) Link speed.

[Table 4-1](#) defines the PEX 8619 Port and Lane configurations. The Lanes are assigned to each enabled Port, in sequence. Ports that are not configured nor enabled are invisible to software.

**Table 4-1. Port Configurations**

<b>STRAP_PORT CFG[3:0]</b>	<b>Port 0</b>	<b>Port 1</b>	<b>Port 2</b>	<b>Port 3</b>	<b>Port 4</b>	<b>Port 5</b>	<b>Port 6</b>	<b>Port 7</b>	<b>Port 8</b>	<b>Port 9</b>	<b>Port 10</b>	<b>Port 11</b>	<b>Port 12</b>	<b>Port 13</b>	<b>Port 14</b>	<b>Port 15</b>
(default) 0h	x1	x1	x1	x1	x1											
(Lane)	0	8	4	12	1	9	2	10	3	11	5	13	6	14	7	15
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	
(Lane)	0-3	8	4	12		9		10		11	5	13	6	14	7	15
2h	x4	x4	x1	x1						x1	x1	x1	x1	x1	x1	
(Lane)	0-3	8-11	4	12							5	13	6	14	7	15
3h	x4	x4	x4	x1								x1		x1		x1
(Lane)	0-3	8-11	4-7	12								13		14		15
4h	x4	x4	x4	x4												
(Lane)	0-3	8-11	4-7	12-15												
5h	x8	x1		x1		x1		x1								
(Lane)	0-7	8		12		9		10		11		13		14		15
6h	x8	x4		x1								x1		x1		x1
(Lane)	0-7	8-11		12								13		14		15
7h	x8	x4		x4												
(Lane)	0-7	8-11		12-15												
8h	x8	x8														
(Lane)	0-7	8-15														

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

#### 4.1.1.2 Port Numbering

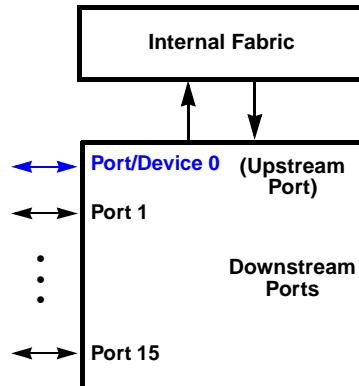
The Port Numbers are 0 through 15, as defined in [Table 4-1](#) and illustrated in [Figure 4-2](#). (Refer to [Section 4.1.1.1](#).)

The Port Numbers have a direct relationship to the downstream Ports for the PCI Device Number assigned to the internal PCI-to-PCI bridges on the internal virtual PCI Bus. *For example*, if Port 1 is a downstream Port, the PCI-to-PCI bridge associated with that Port is Device Number 1. Each downstream Device Number matches its corresponding Port Number. *For example*, if Port 0 is the upstream Port, Ports 1 through 15 are the available downstream Ports, when all Ports are enabled. The Device Numbers for the PCI-to-PCI bridges implemented on the downstream Ports are 1 through 15, respectively.

Any PEX 8619 Port can be configured as, or dynamically changed to be, the upstream Port (Port 0 is recommended). 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 r2.0*.

*Note:* In [Figure 4-2](#), the downstream Port Numbers are only those defined in [Table 4-1](#).

**Figure 4-2. Port Numbering Convention Example (When Port 0 Is Upstream Port)**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 4.2 PCI Express Link Functional Description

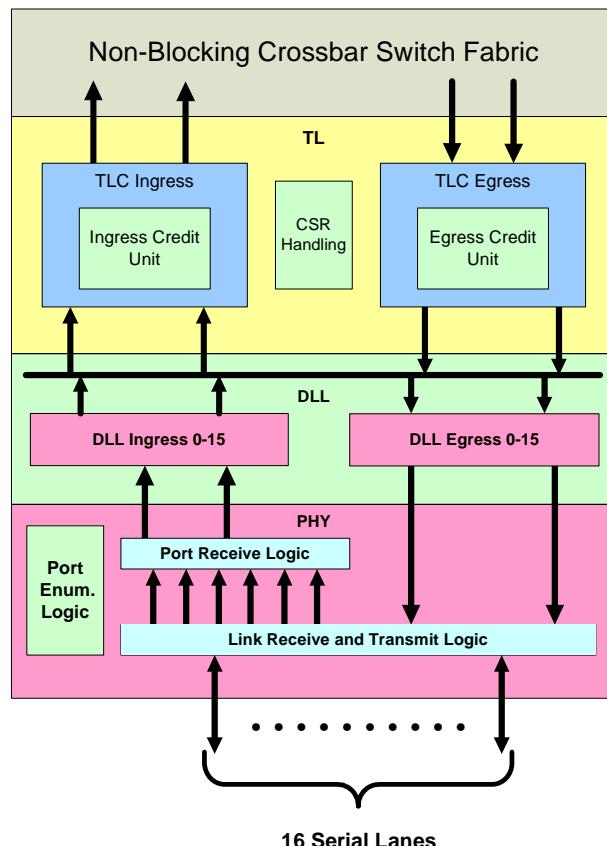
The PEX 8619 groups 16 SerDes together, which can comprise a maximum of 16 Ports. (Refer to [Table 4-1](#).) The Ports forward ingress packets to the internal fabric and central RAM, and pull egress packets from the central RAM to send out of the PEX 8619.

The PEX 8619 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. All 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 is supported through the central internal fabric.

At the top level, the PEX 8619 has a layered organization consisting of the PHY, DLL, and TL blocks, as illustrated in [Figure 4-3](#). 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, 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 4-3. PCI Express Block Diagram**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 4.3 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

### 4.3.1 Physical Layer Features

The PHY module interfaces to the PCI Express Lanes and implements the PHY functions. The quantity of available PCI Express Ports is two, three, four, six, seven, nine, 10, 13, or 16, with a cumulative Lane bandwidth of x16. PHY functions include:

- SerDes modules, which provide all functions required by the *PCI Express Base r2.0*
- User-configurable Port division
- Link widths – x1, x4, or x8; Link width of x2 is also supported, for auto Link-width negotiation with external PCI Express Ports
- Link speeds supported
  - 2.5 GT/s
  - 5.0 GT/s
- Hardware Link training and initialization
- Hardware detection of polarity reversal
- Hardware detection of Lane reversal
- Hardware Autonomous Speed Control supported
- Dynamic Link speed control supported
- Dynamic Link width supported
- Data scrambling/de-scrambling and 8b/10b encode/decode
- Packet framing
- Loopback Master and Slave support
- Programmable test pattern with SKIP Ordered-Set insertion and return data checking
- Receiver error checking (Elastic buffer over/underflow, disparity, and symbol encoding)
- Modified Compliance Pattern support with Receiver Error Counters, per Lane
- Link state Power Management (PM) – Supports L0, L0s, L1, L2/L3 Ready, and L3 Link PM states
- Supports cross-linked upstream Port and downstream Ports

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 4.3.2 Physical Layer Status and Command Registers

The PHY operating conditions are defined in:

- Section 14.14.2, “Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)”
- Section 14.16.14, “Device-Specific Registers – Physical Layer (Offsets B80h – C30h)”
- Section 14.16.16, “Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)”

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

### 4.3.3 Hardware Link Interface Configuration

The PHY can include up to 16 integrated SerDes modules, which 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 r2.0*, as well as the Links (clustered into Ports) that connect the PEX 8619 to other PCI Express devices.

The quantity of enabled Ports, and Link widths associated with those Ports, are configurable, on a Port-by-Port basis. Initial Port configuration is determined by the [STRAP\\_PORTCFG\[3:0\]](#) inputs, serial EEPROM, and/or auto Link-width negotiation. If the PEX 8619 Ports are configured using the auto-negotiation process, the Link widths can narrow or widen (by combining multiple adjacent Lanes).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 4.4 Transaction Layer

The upper layer of the architecture is the Transaction Layer (TL). The TL assembles and disassembles TLPs, which are used to communicate transactions, *such as* Read and Write, as well as certain types of events. The TL also manages credit-based Flow Control (FC) for TLPs.

The TL supports four Address spaces – it includes the three PCI Address spaces (Configuration, Input/Output, and Memory) and adds a Message space. (Refer to [Table 4-2](#).) This specification uses Message space to support all prior sideband signals, *such as* interrupts, Power Management (PM) Requests, and so forth, as in-band Message transactions. PCI Express Message transactions are considered *virtual wires* that support *virtual pins*. As virtual wires, Assert and De-assert Messages are sent when a triggering event changes the wire's state.

**Table 4-2. Address Spaces Support Differing Transaction Types**

Address Space	Transaction Types	Transaction Functions
Configuration	Read/Write	Device configuration or setup
Input/Output		Transfers data from/to an I/O space
Memory		Transfers data from/to a memory location
Message	Baseline/Virtual Wires	General-purpose Messages Event signaling (status, interrupts, and so forth)

All Request packets requiring a Response packet are implemented as Split Transactions. Each packet has a unique identifier that enables Response packets to be directed to the correct originator. The packet format supports various forms of addressing, depending upon the transaction type – *Memory, I/O, Configuration, or Message*.

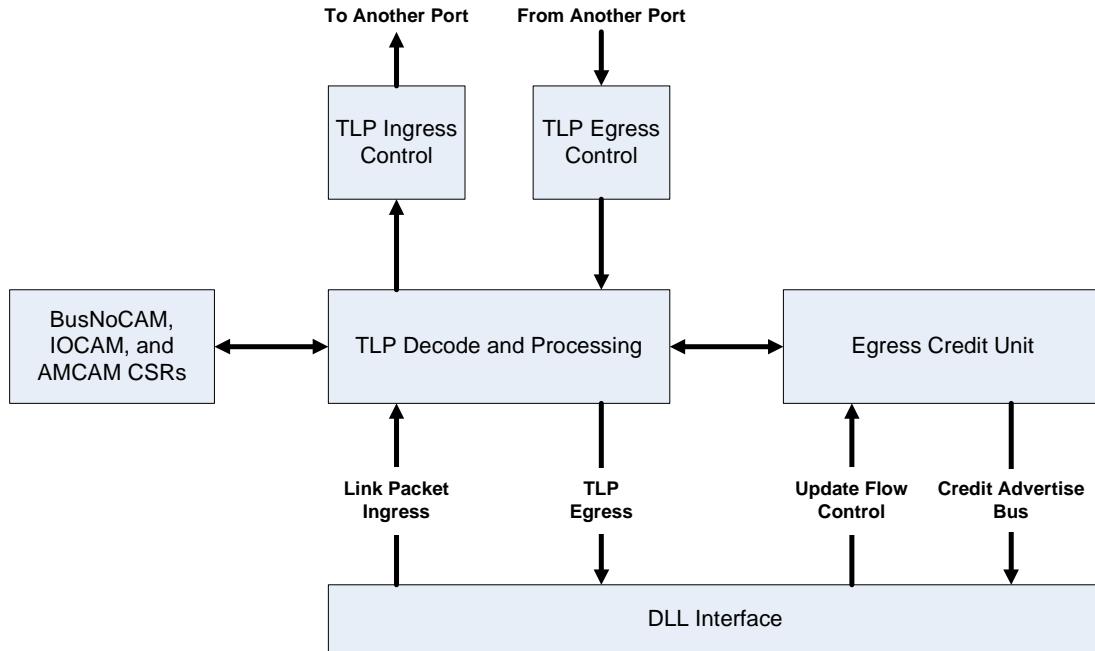
TL functions include:

- Decoding and checking rules for the incoming TLP
- Memory-Mapped Configuration Space register (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
- Shadow CSRs for BusNoCAM/IOCAM/AMCAM/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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

The hardware functions provided by the PEX 8619 to implement the *PCI Express Base r2.0* TL requirements are illustrated in [Figure 4-4](#). The blocks provide a combination of ingress and egress control, as well as the data management at each stage within the flow sequence.

**Figure 4-4. TL Controller**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

#### 4.4.1 Locked Transactions

The PEX 8619 understands Locked transactions; however, 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.

#### 4.4.2 Relaxed Ordering

The PEX 8619 supports Relaxed Ordering for Completions. By default, if the RO attribute is Set within a Completion, then 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 **Ingress Control Shadow** register *No Special Treatment for Relaxed Ordering Traffic* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **664h[5]**).

#### 4.4.3 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 8619 checks the ECRC on all incoming TLPs, if enabled (Port’s **Advanced Error Capabilities and Control** register *ECRC Check Enable* bit (offset FCCh[8]) is Set), 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 8619 can optionally append ECRC to the end of internally generated TLPs, such as Interrupt and Error Messages, if enabled (Port’s **Advanced Error Capabilities and Control** register *ECRC Generation Enable* bit, offset FCCh[6], is Set).

#### 4.4.4 TL Receive/Ingress Protocol

The ingress side TL collects and stores inbound TLP traffic in the packet RAM. The incoming data is checked for ECRC errors, valid type field, length matching the Header *Transfer Size* field, and other TLP-specific errors defined by the *PCI Express Base r2.0*.

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.

#### 4.4.5 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. When enabled, the TL initiates flow credit initialization for VC1, following VC0 initialization.

#### 4.4.6 Flow Control Protocol

The PEX 8619 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 8619

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 8619.

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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 4.5 PCI-Compatible Software Model

The PEX 8619 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 bus. (Refer to Figure 4-5.) PCI-to-PCI bridges are compliant with the PCI and PCI Express system models. Figure 4-5 illustrates the concept of hierarchical PCI-to-PCI bridges, with the bus in the middle being the internal virtual PCI Bus. 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 captures the Type 0 Configuration Write Target Bus Number and Device Number. The upstream Port uses 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.

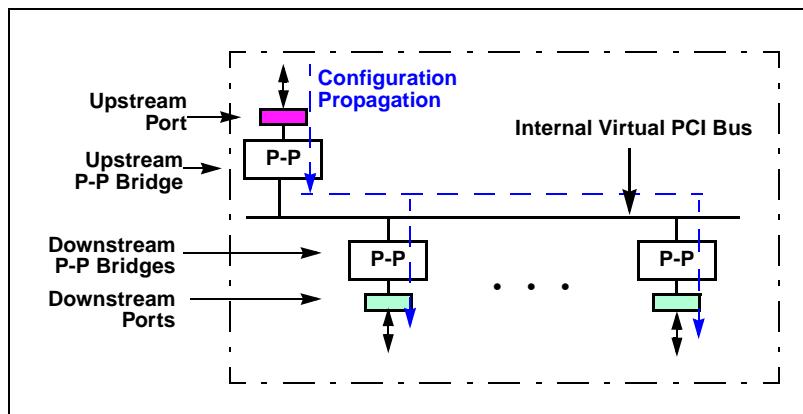
The CSRs in the downstream Port PCI-to-PCI bridges are accessible by Type 1 Configuration Requests received at the upstream Port that target the internal virtual PCI Bus, by having a Bus Number value that matches the upstream bridge's Secondary Bus Number value. Each downstream bridge is associated with a unique Device Number, as explained in [Section 4.1.1.2](#).

The CSRs of downstream devices are hit in two ways. If the Configuration Request matches the PEX 8619 downstream Port Secondary Bus Number, the PEX 8619 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 8619, unchanged. A Type 1 Configuration Request that targets a Bus Number that is not within range is invalid, and is terminated by the PEX 8619 upstream Port as an Unsupported Request (UR).

After all PCI devices have been located and each assigned a Bus Number and Device Number, software can assign a Memory map and I/O map. Requests (Memory or I/O) go downstream if they fall within a bridge's Base and Limit range. In the PEX 8619, each downstream bridge has its own Base and Limit. 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.

**Figure 4-5. PEX 8619 System Configuration Propagation**



THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 5 Reset and Initialization

### 5.1 Reset

This section describes the resets that the PEX 8619 supports. (Refer to [Table 5-1](#).) **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.

**Table 5-1. Reset Summary**

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>	PEX_PERST# 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 Half-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 <b>Bridge Control</b> register <i>Secondary Bus Reset</i> 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 DL_Down behavior, and TLP Requests to that Port are dropped</li><li>• Upstream Port and downstream Ports drain traffic, corresponding to the DL_Down 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 <b>Bridge Control</b> register <i>Secondary Bus Reset</i> 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 DL_Down behavior, and drops TLP Requests that target downstream Ports</li><li>• Upstream Port TL exhibits DL_Up behavior</li></ul>	Initializes downstream Ports registers to default values

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 5.1.1 Fundamental Reset

Fundamental Reset is a hardware mechanism defined by the *PCI Express Base r2.0*, Section 6.6. Fundamental Reset input, through the **PEX\_PERST#** signal, resets all Port states and Configuration registers to default conditions. Reset remains asserted, until the Port's bit is Cleared.

## 5.1.2 Hot Reset

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 (except Port Configuration registers) from serial EEPROM, if present<sup>a</sup>. 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]).

## 5.1.3 Secondary Bus Reset

Any virtual upstream or downstream PCI-to-PCI bridge within the PEX 8619 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 r2.0*. 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 r2.0*, 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 r2.0*, and generates an in-band Hot Reset onto its downstream Link. The registers of that downstream Port are not affected.

---

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 **Serial EEPROM Status** register Status Data from Serial EEPROM fields (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 260h[31:24]). Serial EEPROM presence is reported in the register's **EepPrsnt[1:0]** field [17:16]; a value of 01b or 11b indicates that the serial EEPROM is present.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 5.1.4 Register Bits that Affect Hot Reset

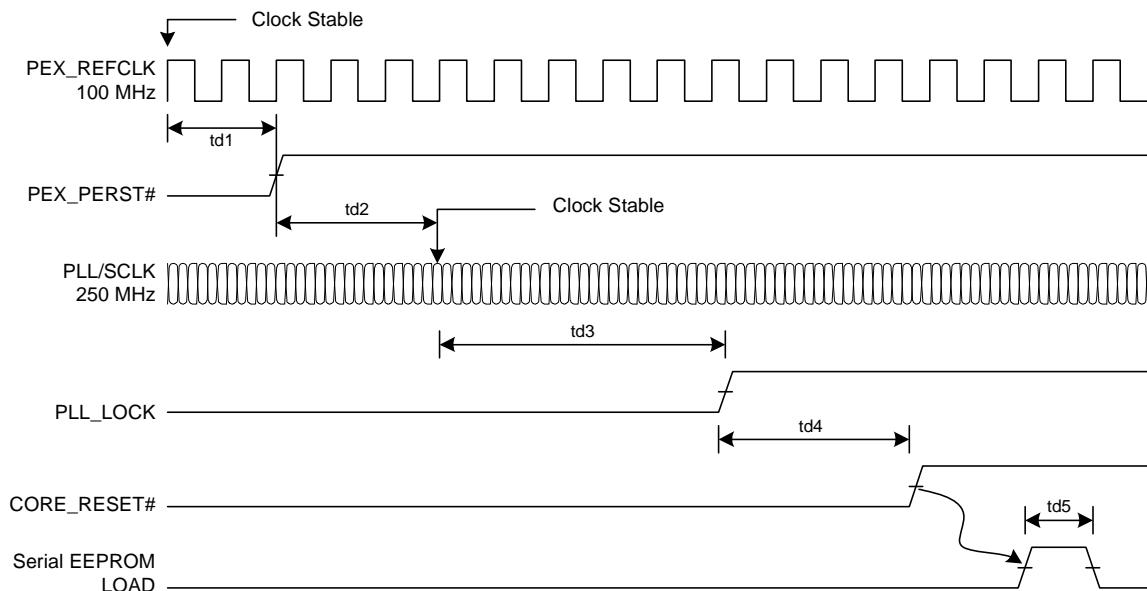
Setting the **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]) generates a Hot Reset to downstream Ports and downstream devices.

## 5.1.5 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 µs
td2	PEX_Reset release to Reset de-bounce	1.32 ms
td3	Reset de-bounce to Phase-Locked Loop (PLL) Lock	105 µs
td4	Reset de-bounce to Core Reset release	2.63 ms
td5	Serial EEPROM load time with no serial EEPROM present	17 µs

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



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 5.2 Initialization Procedure

The PEX 8619 initialization process starts upon exit from a Fundamental Reset. There are two or more steps in the process, depending upon the availability of an external initialization serial EEPROM and I<sup>2</sup>C.

*The initialization sequence executed is as follows:*

1. PEX 8619 reads the Strapping inputs, to determine the upstream Port (**STRAP\_UPSTRM\_PORTSEL[3:0]**) and Lane configuration (**STRAP\_PORTCFG[3:0]**) of each enabled Port.
2. If a serial EEPROM is present, serial EEPROM data is downloaded to the PEX 8619 Configuration registers. The configuration from the Strapping inputs can be changed by serial EEPROM data.

Alternatively, I<sup>2</sup>C can be used to program all the registers (the same as would be done with the serial EEPROM), except, if 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. Because I<sup>2</sup>C is relatively slow, the Links are usually up by the time the first I<sup>2</sup>C Write occurs. The first I<sup>2</sup>C command might be to block system access while the configuration is being changed, by disabling the upstream Port; Ports can be disabled by Setting the Port's **Even/Odd Port Disable** register *Disable Port x* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset **230h[7:0]** and Odd Ports, **234h[7:0]**, respectively).

Switch configuration, including **Port Configuration** (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **574h[3:0]**) and/or upstream Port designation (**Debug Control** register **Upstream Port ID** field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[11:8]**)) can be changed:

- At runtime, by software, if the **Debug Control** register *Hardware/Software Configuration Mode Control* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[15]**) is Set
- By I<sup>2</sup>C and/or serial EEPROM reload

If the **Debug Control** register *Upstream Port ID* field value is to be changed, the register must be written twice, with *Factory Test Only* bit **7** Set in the first Write, and Cleared in the second Write. Changes take effect upon subsequent Hot Reset.

**Note:** As described in *Chapter 7, “I<sup>2</sup>C/SMBus Slave Interface Operation,”* an external I<sup>2</sup>C Master can send the register Read/Write Requests to PEX 8619 after reset. To prevent conflict, first disable the upstream Port, by Setting the Port's *Disable Port x* bit. Restoration of the upstream Port should be the last register Write of the entire I<sup>2</sup>C programming procedure.

3. After configuration from the Strapping inputs, serial EEPROM, and/or I<sup>2</sup>C is complete, the PHY of the configured Ports attempts to bring up the Links. After both components on a Link enter the initial Link Training state, the components proceed through PHY Link initialization and then through Flow Control initialization for VC0 and VC1, preparing the DLL and TL to use the Link. following Flow Control initialization for VC0 and VC1, it is possible for VC0 and VC1 TLPs and DLL Packets (DLLPs) to be transmitted across the Link.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 5.2.1 Default Port Configuration

The default upstream Port selection and overall Link width configuration is determined by Strapping inputs, which must be pulled or tied High to **VDD25** or Low to Ground (**VSS**), to define the default device configuration. (Refer to [Section 3.4.4, “Strapping Signals.”](#)) The configuration defined by the Strapping inputs can be changed by downloading serial EEPROM data, and/or by I<sup>2</sup>C programming followed by a Hot Reset.

### 5.2.2 Default Register Initialization

Each PEX 8619 Port defined in the Port Configuration process has a set of assigned registers that control Port activities and status during standard operation. These registers are programmed to default/initial values, as defined in:

- [Chapter 14, “Transparent Port Registers”](#)
- [Chapter 16, “NT Port Virtual Interface Registers – NT Mode Only”](#)
- [Chapter 17, “NT Port Link Interface Registers – NT Mode Only”](#)
- [Chapter 18, “DMA Interface Registers – DMA Mode Only”](#)

Following a Fundamental Reset, the basic PCI Express Support registers are initially programmed to the values specified in the *PCI Express Base r2.0*. The Device-Specific registers are programmed to the values specified in their register description tables. These registers can be changed by loading new data with the attached serial EEPROM, the I<sup>2</sup>C Slave interface, and/or by CSR accesses using Configuration or Memory Writes; however, registers identified as Read-Only (RO) *cannot* be modified by Configuration nor Memory Write Requests.

The Transparent Ports and NT Port support the following mechanisms for accessing registers by way of the TL, as described in:

- [Section 14.4.1, “PCI r3.0-Compatible Configuration Mechanism”](#)
- [Section 14.4.2, “PCI Express Enhanced Configuration Access Mechanism”](#)
- [Section 14.4.3, “Device-Specific Memory-Mapped Configuration Mechanism”](#)

For NT mode, refer also to [Section 16.3.3](#) or [Section 17.3.3, “Device-Specific Cursor Mechanism.”](#)

### 5.2.3 Device-Specific Registers

The Device-Specific registers are unique to the PEX 8619, and are not referenced in the *PCI Express Base r2.0*. The registers are organized into the following sections:

- [Section 14.14, “Device-Specific Registers \(Offsets 1C0h – 51Ch\)”](#)
- [Section 14.16, “Device-Specific Registers \(Offsets 530h – F8Ch\)”](#)

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

## 5.2.4 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, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, 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  $(16 + 40 + 48) \times 50 \mu\text{s}$  (5,200  $\mu\text{s}$ ).

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, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, 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 2.5 to 3.3V). 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 r2.0* 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.

For further details, refer to [Chapter 6, “Serial EEPROM Controller.”](#)

## 5.2.5 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, “I<sup>2</sup>C Slave Interface.”](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

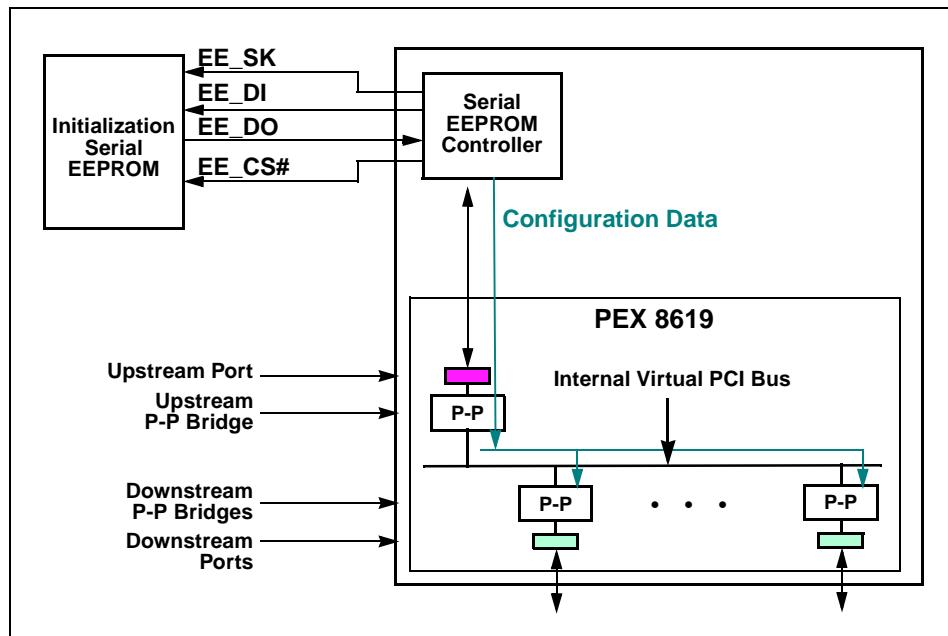
## 6.1 Overview

The PEX 8619 provides a Serial EEPROM Controller and interface to Serial Peripheral Interface (SPI)-compatible serial EEPROMs, as illustrated in [Figure 6-1](#). This interface consists of a Chip Select, Clock and Write Data outputs, and a Read Data input, and operates at a programmable frequency of up to 17.86 MHz. The PEX 8619 supports serial EEPROMs that use 1-, 2-, or 3-byte addressing (2-byte addressing is recommended); the PEX 8619 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 8619, immediately after reset, the Serial EEPROM Controller reads data from the serial EEPROM that is used to update the PEX 8619 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 Expansion ROM (starting at the Expansion ROM Base Address), the PEX 8619 reads from the serial EEPROM, to return the requested ROM image.

**Figure 6-1. Serial EEPROM Connections**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 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
- 4-byte Write/Read access to the serial EEPROM, through the upstream Port
- Serial EEPROM data format allows for loading registers by 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
- No Cyclic Redundancy Check (CRC), single *Valid* byte at start of serial EEPROM memory
- Supports Expansion ROM for the NT Port (*not supported* for 1-byte address serial EEPROMs)

## 6.3 Serial EEPROM Load following Upstream Port Reset

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

- 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 occur:
  - **PEX\_PERST#** is returned High, following a Fundamental Reset (*such as a Cold or Warm Reset to the entire chip*)
  - Hot Reset is received at the upstream Port (downloading upon this event can be optionally disabled, by Setting the **Debug Control** register *Disable Serial EEPROM Load on Hot Reset* and/or *Upstream Hot Reset Control* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[17]** and/or **16**, respectively))
  - Upstream Port exits a *DL\_Down* state (downloading upon this event can be optionally disabled, by Setting the **Debug Control** register *Upstream Port and NT-Link DL\_Down Reset Propagation Disable* and/or *Upstream Hot Reset Control* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[20]** and/or **16**, 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 **Serial EEPROM Status** register Status Data from Serial EEPROM fields (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **260h[31:24]**). Serial EEPROM presence is reported in the register's **EepPrsnt[1:0]** field [17:16]; a value of 01b or 11b indicates that the serial EEPROM is present.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 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 8619 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 8619 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 8619 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 4 register offset 1F8h, shift the address to the right by 2 bits (this becomes 07Eh) and concatenate 0001\_00b. The resulting DWord address in the serial EEPROM will be 0001\_0000\_0111\_1110b, which is 107Eh.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

Location	Value	Description
0h	5Ah	Validation Signature
1h	00h	<b>Reserved</b>
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)
...	...	...
FFFFh	REGDATA (Byte 3)	Last Configuration Register Data (Byte 3)

**Note:** The first Configuration register programmed by the serial EEPROM must be the **Debug Control** register (Port 0, offset **1DCh**), serial EEPROM locations 4h through 9h, as listed in **Table 6-1**.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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

Port Number	REGADDR Bits [15:10] Value <sup>a</sup>	Port Identifier
Port 0	0000_00b	0000h
Port 1	0000_01b	0400h
Port 2	0000_10b	0800h
Port 3	0000_11b	0C00h
Port 4	0001_00b	1000h
Port 5	0001_01b	1400h
Port 6	0001_10b	1800h
Port 7	0001_11b	1C00h
Port 8	0010_00b	2000h
Port 9	0010_01b	2400h
Port 10	0010_10b	2800h
Port 11	0010_11b	2C00h
Port 12	0011_00b	3000h
Port 13	0011_01b	3400h
Port 14	0011_10b	3800h
Port 15	0011_11b	3C00h
NT Port Link Interface	1100_00b	C000h
NT Port Virtual Interface <sup>b</sup>	00XX_XXb	XX00h
NT PCI-to-PCI Bridge	1100_01b	C400h
DMA CSRs	1100_10b	C800h
DMA Descriptor RAM	1100_11b	CC00h

- a. *Encodings not listed are reserved.*
- b. *For NT Port Virtual Interface registers, use the value for the Port Number that is configured as the NT Port (as designated by the [STRAP\\_NT\\_UPSTRM\\_PORTSEL\[3:0\]](#) inputs or [Debug Control](#) register NT Port Number field (Port 0, offset [IDCh\[27:24\]](#))).*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 6.5 Serial EEPROM Initialization

After the device Reset is de-asserted, the PEX 8619 determines whether a serial EEPROM is present. 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 **Serial EEPROM Status** register *Status Data from Serial EEPROM* fields (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 260h[31:24]). Serial EEPROM presence is reported in the register's *EepPrsnt[1:0]* field [17:16]; a value of 01b or 11b indicates that the serial EEPROM is present. A pull-up resistor on the **EE\_DO** input produces a value of FFh if a serial EEPROM is not installed.

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 8619. 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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, 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 8619 Serial EEPROM registers) default to a serial EEPROM address width of 1 byte unless the **Serial EEPROM Status** register *EepAddrWidth Override* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, 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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 268h[2:0]). The default clock frequency is 1 MHz. At this clock rate, it takes approximately 48 µs 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 6.6 PCI Express Configuration, Control, and Status Registers

The PCI Express Configuration, Control, and Status registers that can be initialized are detailed in:

- Chapter 14, “Transparent Port Registers”
- Chapter 16, “NT Port Virtual Interface Registers – NT Mode Only”
- Chapter 17, “NT Port Link Interface Registers – NT Mode Only”
- Chapter 18, “DMA Interface Registers – DMA Mode Only”

## 6.7 Serial EEPROM Registers

The Serial EEPROM register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 260h through 26Ch) parameters defined in [Section 14.14.2, “Device-Specific Registers – Physical Layer \(Offsets 200h – 25Ch\),”](#) can be changed, using the serial EEPROM. It is recommended that the first serial EEPROM entry (after the **Debug Control** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh) entry, if programmed), 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 (WRSR) command, to enable the Write Protection feature(s) within the serial EEPROM data, if needed.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 6.8 Serial EEPROM Random Write/Read Access

To access the serial EEPROM, a PCI Express, I<sup>2</sup>C, and/or SMBus Master uses the following registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port):

- **Serial EEPROM Status and Control** (offset 260h)
- **Serial EEPROM Buffer** (offset 264h)
- **Serial EEPROM 3<sup>rd</sup> 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).

### 6.8.1 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 3<sup>rd</sup> 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 **Serial EEPROM Control** register *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 0. At this time, another serial EEPROM access can be started.

Because each PEX 8619 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 8619 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).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 6.8.2 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 3<sup>rd</sup> Address Byte** register *Serial EEPROM 3<sup>rd</sup> 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 the *EepCmd[2:0]* field [15:13], together with the serial EEPROM address. Serial EEPROM Address bits [14:2] must be programmed into the **Serial EEPROM Control** register *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).
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.

## 6.8.3 Programming a Blank Serial EEPROM

The PEX 8619 supports 1-, 2-, or 3-byte serial EEPROM addressing. 8-Kbit to 512-Kbit SPI EEPROMs use 2-byte addressing. The PEX 8619 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 (*such as* if the first DWord is programmed to the value 5A00\_FFFFh), 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, program to 0000\_005Ah.

A 2-byte address serial EEPROM that is blank (or corrupted) can be programmed according to the following procedure (when the PEX 8619 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 8619 so that it re-detects the serial EEPROM.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

The **Debug Control** register *Load Only EEPROM NT-Link on Hot Reset* and *Inhibit EEPROM NT-Link Load on Hot Reset* bits (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[31:30]**, 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 (Offset 1DCh[31:30] Values)**

Bit 31 Value	Bit 30 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.

## 6.10 NT Port Expansion ROM – NT Mode Only

The PEX 8619 NT Port Virtual and Link Interfaces support Expansion ROM, as defined in the *PCI r3.0*. Expansion ROM can be implemented for either Port, 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 (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **268h[16]**) value. When the 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 Control** register *Expansion ROM Virtual Side* bit (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **660h[23]**). 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 Expansion ROM's location in the serial EEPROM is programmed in the **Serial EEPROM 3rd Address Byte** register *Expansion ROM Base Address* field (NT Port Virtual Interface if Port 0 is a Legacy NT 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 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 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 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



# 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\\_ADDR\[2:0\]](#), [I2C\\_SCL0](#), and [I2C\\_SDA0](#) signals for PEX 8619 register access by an I<sup>2</sup>C Master. The [I2C\\_SCL1](#) and [I2C\\_SCL1](#) signals form the PEX 8619 I<sup>2</sup>C Master interface, which is used only for Serial Hot Plug operation. (Refer to [Chapter 12, “Hot Plug Support.”](#))*

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 [I<sup>2</sup>C Bus, v2.1](#).

The PEX 8619 is an I<sup>2</sup>C Slave. Slave operations allow the PEX 8619 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 8619 registers through the I<sup>2</sup>C Slave interface. I<sup>2</sup>C provides an alternative to using a serial EEPROM. I<sup>2</sup>C can also be used for debugging, such as if the PEX 8619 upstream Port fails to linkup.

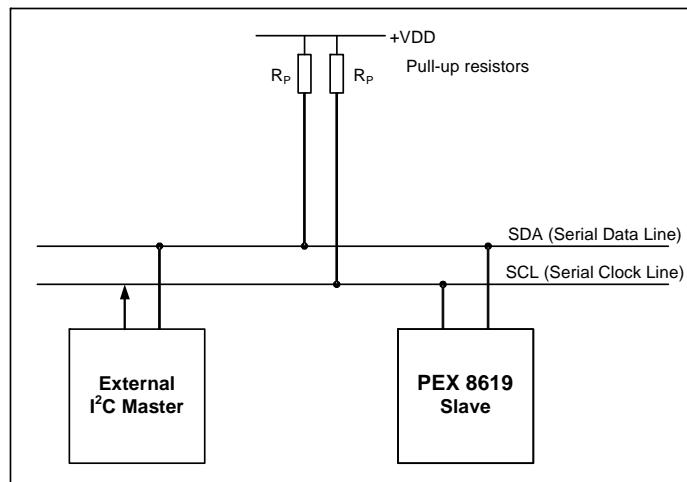
Accordingly, it is recommended that both I<sup>2</sup>C/SMBus access, and the serial EEPROM (or at least its footprint), be included in designs.

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 8619 registers, using an Aardvark USB-I<sup>2</sup>C adapter connected to this header. (Refer to the *PEX 8619 RDK Hardware Reference Manual* for the header pin design.)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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

To access the PEX 8619 Configuration registers through the I<sup>2</sup>C Slave interface, the PEX 8619 I<sup>2</sup>C Slave address must be configured.

The PEX 8619 supports a 7-bit I<sup>2</sup>C Slave address. The 7-bit I<sup>2</sup>C Address bits can be configured by the serial EEPROM (recommended, if the default address must be changed), or by a Memory Write, in the **I<sup>2</sup>C Configuration** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **294h**, default value **38h**), with the lower three bits of the address derived from the **I2C\_ADDR[2:0]** inputs. Bits [6:0] correspond to Address Byte bits [7:1], with bit 0 of the byte indicating a Write (0) or Read (1).

The I2C\_ADDR[2:0] inputs can be pulled or tied High or Low, to select a different Slave address. Up to eight PEX 8619 devices can share the same I<sup>2</sup>C Bus segment without conflict, provided that each PEX 8619 has its I2C\_ADDR[2:0] inputs strapped to a unique state. More than eight PEX 8619 devices can share the I<sup>2</sup>C Bus, however, if the upper Address bits are programmed in the serial EEPROM.

## 7.2.3 I<sup>2</sup>C Slave Interface Register

The **I<sup>2</sup>C Slave Interface** register, **I<sup>2</sup>C Configuration**, is described in [Section 14.14.5, “Device-Specific Registers – I<sup>2</sup>C Slave Interface \(Offsets 290h – 2C4h\).”](#) The default I<sup>2</sup>C Slave address can be changed in the **I<sup>2</sup>C Configuration** register to a different value, using the serial EEPROM or a Memory Write.

**The I<sup>2</sup>C Slave address must not be changed by an I<sup>2</sup>C Write command.** (Refer to [Section 7.2.2.](#))

Other I<sup>2</sup>C Slave interface registers exist; however, they are for **Factory Test Only**.

## 7.2.4 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 Command Phase 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 8619 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 8619, 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 8619, 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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

The PEX 8619 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 8619 register is not modified.

The PEX 8619 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 8619 protocol), the PEX 8619 returns a NAK for the extra Data byte(s). (For further details regarding I<sup>2</sup>C protocol, refer to the [I<sup>2</sup>C 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

Byte	Bit(s)	Description
1 <sup>st</sup> (0)	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 011b = Write register Do not use other encodings for Writes.
2 <sup>nd</sup> (1)	7:4	<b>Reserved</b> Should be Cleared.
	3:0	<b>Port Selector, Bits [4:1]</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector.
3 <sup>rd</sup> (2)	7	<b>Port Selector, Bit 0</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector. <i>Port Selector</i> , bits [4:0] select the Port to access. 00h – 0Fh for 16 Ports (including NT Port Virtual Interface, if NT mode is enabled) 10h for NT Port Link Interface 11h for NT PCI-to-PCI Bridge 12h for DMA function 13h for DMA Descriptor RAM 14h – 1Fh are <i>reserved</i> .
	6	<b>Reserved</b> Should be Cleared.
	5:2	<b>Byte Enables</b> <b>Bit      Description</b> 2      Byte Enable for Byte 0 (PEX 8619 register bits [7:0]) 3      Byte Enable for Byte 1 (PEX 8619 register bits [15:8]) 4      Byte Enable for Byte 2 (PEX 8619 register bits [23:16]) 5      Byte Enable for Byte 3 (PEX 8619 register bits [31:24])  0 = Corresponding PEX 8619 register byte will not be modified 1 = Corresponding PEX 8619 register byte will be modified  All 16 combinations are valid values.
	1:0	<b>PEX 8619 Register Address [11:10]</b>
4 <sup>th</sup> (3)	7:0	<b>PEX 8619 Register Address [9:2]</b> <i>Note:</i> 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 7.2.5.1 I<sup>2</sup>C Register Write

The following tables illustrate a sample I<sup>2</sup>C packet for writing the PEX 8619 **MSI Upper Address** register (offset **50h**) for Port 4, with data **1234\_5678h**.

*Note: The PEX 8619 has a default I<sup>2</sup>C Slave address [6:0] value of **38h**, with the **I2C\_ADDR[2:0]** inputs having a value of 000. 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 7-3. I<sup>2</sup>C Register Write Access Example – 1<sup>st</sup> Cycle**

Phase	Value	Description
Address	70h	<b>Bits [7:1] for PEX 8619 I<sup>2</sup>C Slave Address (38h)</b> Last bit (bit 0) for Write = 0.

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

Byte	Value	Description	
0	03h	[7:3]	<b>Reserved</b> Should be Cleared.
		[2:0]	<b>Command</b> 011b = Write register
1	00h for Port 1 01h for Port 2 01h for Port 3	[7:4]	<b>Reserved</b> Should be Cleared.
		[3:0]	<b>Port Selector, Bits [4:1]</b>
2	BCh for Port 1 3Ch for Port 2 BCh for Port 3	7 6 [5:2]	<b>Port Selector, Bit 0</b> <b>Reserved</b> Should be Cleared. <b>Byte Enables</b> All active.
		[1:0]	<b>PEX 8619 Register Address, Bits [11:10]</b>
3	14h	[7:0]	<b>PEX 8619 Register Address, Bits [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 0111_000b	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_0010b	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

When the I<sup>2</sup>C Master attempts to read a PEX 8619 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 [I<sup>2</sup>C 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 8619 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 8619 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](#).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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

<b>Byte</b>	<b>Bit(s)</b>	<b>Description</b>
1 <sup>st</sup> (0)	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 100b = Read register Do not use other encodings for Reads.
2 <sup>nd</sup> (1)	7:4	<b>Reserved</b> Should be Cleared.
	3:0	<b>Port Selector, Bits [4:1]</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector.
3 <sup>rd</sup> (2)	7	<b>Port Selector, Bit 0</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector. <i>Port Selector</i> , bits [4:0] select the Port to access. 00h – 0Fh for 16 Ports (including NT Port Virtual Interface, if NT mode is enabled) 10h for NT Port Link Interface 11h for NT PCI-to-PCI Bridge 12h for DMA function 13h for DMA Descriptor RAM 14h – 1Fh are <i>reserved</i> .
	6	<b>Reserved</b> Should be Cleared.
	5:2	<b>Byte Enables</b> <b>Bit</b> <b>Description</b> 2      Byte Enable for Byte 0 (PEX 8619 register bits [7:0]) 3      Byte Enable for Byte 1 (PEX 8619 register bits [15:8]) 4      Byte Enable for Byte 2 (PEX 8619 register bits [23:16]) 5      Byte Enable for Byte 3 (PEX 8619 register bits [31:24])  0 = Corresponding PEX 8619 register byte will not be modified 1 = Corresponding PEX 8619 register byte will be modified  All 16 combinations are valid values.
	1:0	<b>PEX 8619 Register Address [11:10]</b>
4 <sup>th</sup> (3)	7:0	<b>PEX 8619 Register Address [9:2]</b> <i>Note:</i> 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**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	STOP
Command Byte 0	A	Command Byte 1	A	Command Byte 2	A	Command Byte 3	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 7.2.6.1 I<sup>2</sup>C Register Read Address Phase and Command Packet

The following is a sample I<sup>2</sup>C packet for reading the PEX 8619 **Serial EEPROM Buffer** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **264h**) for Port 4, assuming the register value is ABCD\_EF01h.

*Note: The PEX 8619 has a default I<sup>2</sup>C Slave address [6:0] value of **38h**, with the **I2C\_ADDR[2:0]** inputs having a value of 000. 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	Description
Address	70h	<b>Bits [7:1] for PEX 8619 I<sup>2</sup>C Slave Address (38h)</b> Last bit (bit 0) for Write = 0.

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

Byte	Value	Description	
0	04h	[7:3]	<b>Reserved</b> Should be Cleared.
		[2:0]	<b>Command</b> 100b = Read register
1	00h for Port 1 01h for Port 2 01h for Port 3	[7:4]	<b>Reserved</b> Should be Cleared.
		[3:0]	<b>Port Selector, Bits [4:1]</b>
2	BCh for Port 1 3Ch for Port 2 BCh for Port 3	7 6 [5:2]	<b>Port Selector, Bit 0</b> <b>Reserved</b> Should be Cleared. <b>Byte Enables</b> All active. <b>PEX 8619 Register Address, Bits [11:10]</b>
3	99h	[1:0]	<b>PEX 8619 Register Address, Bits [9:2]</b>
		[7:0]	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 7.2.6.2 I<sup>2</sup>C Register Read Data Packet

**Note:** The PEX 8619 has a default I<sup>2</sup>C Slave address [6:0] value of 38h, with the I<sub>2</sub>C\_ADDR[2:0] inputs having a value of 000. 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 7-9. I<sup>2</sup>C Register Read Access Example – 1<sup>st</sup> Cycle**

Phase	Value	Description
Address	71h	Bits [7:1] for PEX 8619 I <sup>2</sup> C Slave Address (38h) Last bit (bit 0) for Read = 1.
Read	ABh	<b>Byte 3 of Register Read</b>
	CDh	<b>Byte 2 of Register Read</b>
	EFh	<b>Byte 1 of Register Read</b>
	01h	<b>Byte 0 of Register Read</b>

**Figure 7-6. 1<sup>st</sup> Packet – I<sup>2</sup>C Command Phase**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address 0111_000b	Read/Write Bit 0 = Write	A

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	STOP
Command Byte 0 0000_0100b	A	Command Byte 1 0000_0010b	A	Command Byte 2 0011_1110b	A	Command Byte 3 1001_1001b	P

**Figure 7-7. 2<sup>nd</sup> Packet – I<sup>2</sup>C Read Phase**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address[7:1] 0111_000b	Read/Write Bit 1 = Read	A

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	STOP
Register Byte 3 1010_1011b	A	Register Byte 2 1100_1101b	A	Register Byte 1 1110_1111b	A	Register Byte 0 0000_0001b	P

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 7.3 SMBus Slave Interface

### 7.3.1 SMBus Features

- Compliant to the *SMBus v2.0*
- Supports the SMBus Slave function only
- PEX 8619 internal registers can be read and written, through the SMBus Slave interface
- Supports Address Resolution Protocol (ARP-capable)
- Strapping inputs, serial EEPROM, software, or ARP Set the Slave address
- Supports Block Read, Block Write, and Block Read - Block Write Process Call commands to access the registers
- Supports Packet Error Checking
- 10 to 100 KHz Bus operation frequency range

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 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 8619 implements an *SMBus v2.0*-compliant Slave device, and is used to read and write PEX 8619 registers, through SMBus commands. The PEX 8619 SMBus uses the same SDA data and SCL clock balls that are used for I<sup>2</sup>C, and the I2C\_ADDR[1:0] inputs, to define address assignment (I2C\_ADDR2 is not used as an Address bit in SMBus mode). At any time, either the I<sup>2</sup>C or SMBus feature is enabled, dependent upon the **SMBus Configuration** register *SMBus Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[0]) state, which is latched (at Fundamental Reset) to the inverse value of the **STRAP\_SMBUS\_EN#** input. Software can toggle this bit to switch between I<sup>2</sup>C and SMBus functionality.

The PEX 8619 SMBus Slave interface supports three command protocols for register access:

- Block Write
- Block Read
- Block Read - Block Write Process Call

The PEX 8619 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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[8]) is Cleared; this bit is initially latched (at Fundamental Reset) to the value of the I2C\_ADDR2 input.

If ARP is disabled, by I2C\_ADDR2 input being pulled or tied High, the SMBus Slave Address bits [6:2] default to value 001\_10b. Address bits [1:0] are initially latched (at Fundamental Reset) to the I2C\_ADDR[1:0] input values, which allow a maximum of four SMBus-enabled PEX 8619s to co-exist on the same SMBus segment. Software can change the SMBus Slave address, by programming the **SMBus Configuration** register *SMBus Device Address* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[7:1]).

The PEX 8619 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 SMBus Commands Supported

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 Read - Block Write Process Call (commands BAh, CDh) can also be used to read registers

SMBus Commands that are not supported by the PEX 8619 (Quick Command, Send Byte, Receive Byte, Write Byte, Write Word, Read Byte, Read Word, and Process Call), are NACKed.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

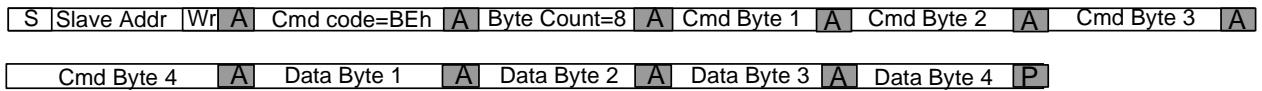
### 7.3.3.1 SMBus Block Write

The Block Write command is used to write to the PEX 8619 registers. General SMBus Block Writes are illustrated in [Figure 7-8](#) and [Figure 7-9](#). The sequence of Bytes include the following, in the sequence listed:

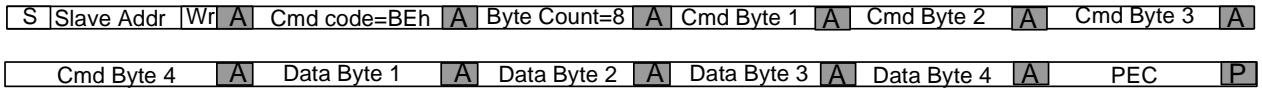
- 7-bit address,
- Command Code that indicates it is Block Write,
- *Byte Count* field with a value of 8h that indicates 4 bytes to set up the register to write (Port Number, register address, Command Byte Enable, and so forth), followed by
- 4 bytes of data to be written into the register

[Figure 7-10](#) explains the elements used in [Figure 7-8](#) and [Figure 7-9](#), and [Figure 7-11](#) indicates the Data Bytes written.

**Figure 7-8. SMBus Block Write Command Format, to Write to a PEX 8619 Register without PEC**



**Figure 7-9. SMBus Block Write Command Format, to Write to a PEX 8619 Register with PEC**



**Figure 7-10. 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 7-11. SMBus Block Write Bytes, as Written to Register**

31:24	23:16	15:8	7:0
Data Byte 1	Data Byte 2	Data Byte 3	Data Byte 4

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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 7-10** 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 8619 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 8619 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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[9]). The Byte Count value, by definition, does not include the PEC byte.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 7-10. 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 100b = Read register All other encodings are <i>reserved. Do not use.</i>
Command Byte 2	7:4	<b>Reserved</b> Should be Cleared.
	3:0	<b>Port Selector, Bits [4:1]</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector.
Command Byte 3	7	<b>Port Selector, Bit 0</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector. <i>Port Selector</i> , bits [4:0] select the Port to access. 00h – 0Fh for 16 Ports (including NT Port Virtual Interface, if NT mode is enabled) 10h for NT Port Link Interface 11h for NT PCI-to-PCI Bridge 12h for DMA function 13h for DMA Descriptor RAM 14h – 1Fh are <i>reserved</i> .
	6	<b>Reserved</b> Should be Cleared.
	5:2	<b>Byte Enables</b> <b>Bit</b> <b>Description</b> 2      Byte Enable for Byte 0 (PEX 8619 register bits [7:0]) 3      Byte Enable for Byte 1 (PEX 8619 register bits [15:8]) 4      Byte Enable for Byte 2 (PEX 8619 register bits [23:16]) 5      Byte Enable for Byte 3 (PEX 8619 register bits [31:24]) 0 = Corresponding PEX 8619 register byte will not be modified 1 = Corresponding PEX 8619 register byte will be modified All 16 combinations are valid values.
Command Byte 4	1:0	<b>PEX 8619 Register Address [11:10]</b>
	7:0	<b>PEX 8619 Register Address [9:2]</b> <i>Note:</i> 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Sample Register Write Byte Sequence Using SMBus Block Write**

An SMBus Block Write packet to write to the **MSI Upper Address** [63:32] register (offset **50h**) in Port 3, is listed in [Table 7-11](#). The register value is 1234\_5678h, with all Bytes enabled, and without PEC. The default SMBus Device Address is 001\_100b.

**Table 7-11. Sample SMBus Block Write Byte Sequence**

Byte Number	Byte Type	Value	Description
1	Address	70h	Bits [7:1] for the PEX 8619 default Slave address of 38h, with bit 0 Cleared to indicate a Write.
2	Command Code	B Eh	Command Code for register Write, using a Block Write.
3	Byte Count	08h	Byte Count. Four Command Bytes and Four Data Bytes.
4	Command Byte 1	03h	For Write command.
5	Command Byte 2	01h	Bits [2:0] – Port Selector [3:1].
6	Command Byte 3	B Ch	Bit 7 is Port Selector LSB. Bit 6 is <i>reserved</i> . Bits [5:2] are the four Byte Enables; all are active. Bits [1:0] are register Address bits [11:10].
7	Command Byte 4	14h	PEX 8619 Register Address bits [9:2] (for offset <b>50h</b> ).
8	Data Byte 1	12h	Data MSB.
9	Data Byte 2	34h	Data Byte for register bits [23:16].
10	Data Byte 3	56h	Data Byte for register bits [15:8].
11	Data Byte 4	78h	Data LSB.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 7.3.3.2 SMBus Block Read

A Block Read command is used to read PEX 8619 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 8619 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 Read - Block Write Process Call. This command is defined by the *SMBus v2.0*, and performs a Block Write and Block Read, using a single command. 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.*

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

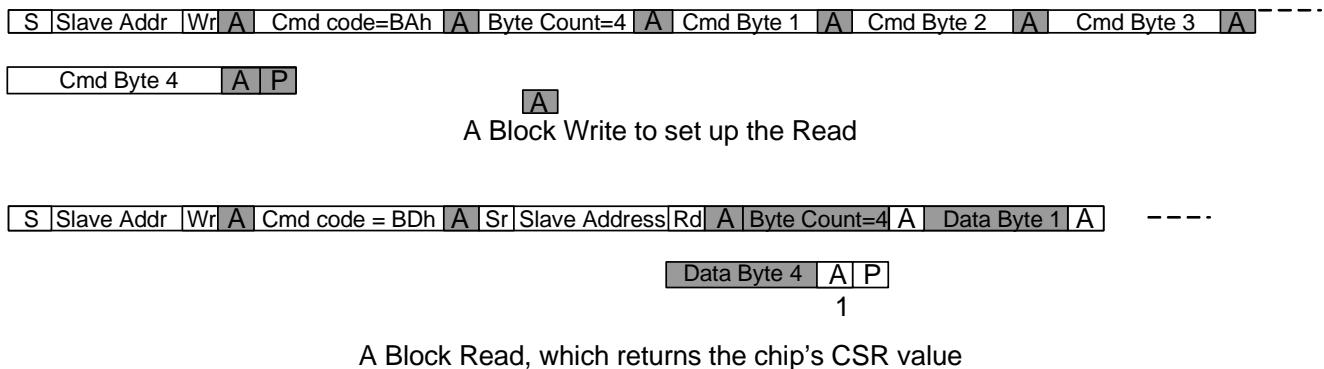
A general SMBus Block Write and Block Read sequence is illustrated in [Figure 7-12](#).

[Table 7-12](#) describes the Byte definitions for a Block Write bus protocol, to prepare for a subsequent Block Read of the PEX 8619 register.

The PEX 8619 always NACKs any incorrect command sequences, starting with the wrong Byte. Upon receiving the Block Read command, the PEX 8619 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 8619 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 8619.

Incorrect command sequences are always NACKed, starting with the byte that is incorrect. (Refer to [Table 7-13](#).) On the Block Read command, a PEC is returned to the Master, if after the 4<sup>th</sup> byte of CSR data, the return Master still requests for one additional byte. As a Slave, the PEX 8619 will know the end of the Master Read cycle, by observing the NACK for the last byte read from the Master.

**Figure 7-12. SMBus Block Write to Set up Read, and Resulting Read that Returns CSR Value**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 7-12. 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> 011b = Write register 100b = Read register All other encodings are <i>reserved. Do not use.</i>
Command Byte 2	7:4	<b>Reserved</b> Should be Cleared.
	3:0	<b>Port Selector, Bits [4:1]</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector.
Command Byte 3	7	<b>Port Selector, Bit 0</b> 2 <sup>nd</sup> Command byte, bits [3:0], and 3 <sup>rd</sup> Command byte, bit 7, combine to form a 5-bit Port Selector. <i>Port Selector</i> , bits [4:0] select the Port to access. 00h – 0Fh for 16 Ports (including NT Port Virtual Interface, if NT mode is enabled) 10h for NT Port Link Interface 11h for NT PCI-to-PCI Bridge 12h for DMA function 13h for DMA Descriptor RAM 14h – 1Fh are <i>reserved</i> .
	6	<b>Reserved</b> Should be Cleared.
	5:2	<b>Byte Enables</b> <b>Bit</b> <b>Description</b> 2      Byte Enable for Byte 0 (PEX 8619 register bits [7:0]) 3      Byte Enable for Byte 1 (PEX 8619 register bits [15:8]) 4      Byte Enable for Byte 2 (PEX 8619 register bits [23:16]) 5      Byte Enable for Byte 3 (PEX 8619 register bits [31:24]) 0 = Corresponding PEX 8619 register byte will not be modified 1 = Corresponding PEX 8619 register byte will be modified All 16 combinations are valid values.
Command Byte 4	1:0	<b>PEX 8619 Register Address [11:10]</b>
	7:0	<b>PEX 8619 Register Address [9:2]</b> <i>Note:</i> 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.

**Table 7-13. 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).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Sample CSR Read Byte Sequence, Using SMBus Block Write Followed by SMBus Block Read**

An SMBus sequence to write and read the **MSI Upper Address** [63:32] register (offset **50h**) in Port 3, is listed in [Table 7-14](#) and [Table 7-15](#), respectively. The register value is ABCD\_EF01h, and without PEC. The Block Write sets up the Port Numbers, Register address and Byte Enables, and the Block Read performs the real Read operation. The default SMBus Device Address is 001\_100b.

**Table 7-14. SMBus Block Write Portion**

Byte Number	Byte Type	Value	Description
1	Address	70h	Bits [7:1] value for the PEX 8619 Slave address of 38h, with bit 0 Cleared to indicate a Write.
2	Block Write Command Code	BAh	Command Code for register Read setup, using a Block Write.
3	Byte Count	04h	Byte Count. Four Command Bytes.
4	Command Byte 1	04h	Write command.
5	Command Byte 2	01h	Bit 3 = 0, <b>Port Selector [3:0]</b> = Port Selector bits (2 <sup>nd</sup> Command byte, bits [3:0] and 3 <sup>rd</sup> Command byte, bit 7).
6	Command Byte 3	BCh	Bit 7 is Port Selector LSB. Bit 6 is <i>reserved</i> . Bits [5:2] are the four Byte Enables; all are active. Bits [1:0] are register Address bits [11:10].
7	Command Byte 4	9Dh	PEX 8619 Register Address bits [9:2] (for offset <b>50h</b> ).

**Table 7-15. SMBus Block Read Portion**

Byte Number	Byte Type	Value	Description
1	Address	70h	Bits [7:1] value for the PEX 8619 Slave address of 38h, with bit 0 Cleared to indicate a Write.
2	Block Read Command Code	BDh	Command code for Block Read of PEX 8619 registers.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 7-16. SMBus Read Command following Repeat START from Master**

Byte Number	Byte Type	Value	Description
1	Address	71h	Bits [7:1] value for the PEX 8619 Slave address of 38h, with bit 0 Set to indicate a Read.

**Table 7-17. PEX 8619 SMBus Return Bytes**

Byte Number	Byte Type	Value	Description
1	Byte Count	04h	Four Bytes in register.
2	Data Byte 1	ABh	Register data MSB.
3	Data Byte 2	CDh	Register data [23:16].
4	Data Byte 3	EFh	Register data [15:8].
5	Data Byte 4	01h	Register data LSB.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

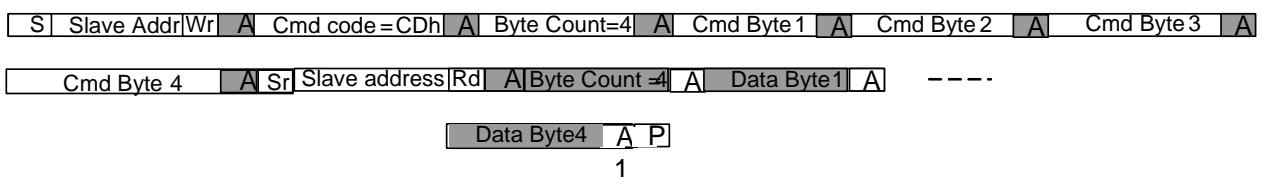
### 7.3.3.3 CSR Read, Using SMBus Block Read - Block Write Process Call

A general SMBus Block Read - Block Write Process Call sequence is illustrated in [Figure 7-13](#). Alternatively, a general SMBus Block Read - Block Write Process Call with PEC sequence is illustrated in [Figure 7-14](#).

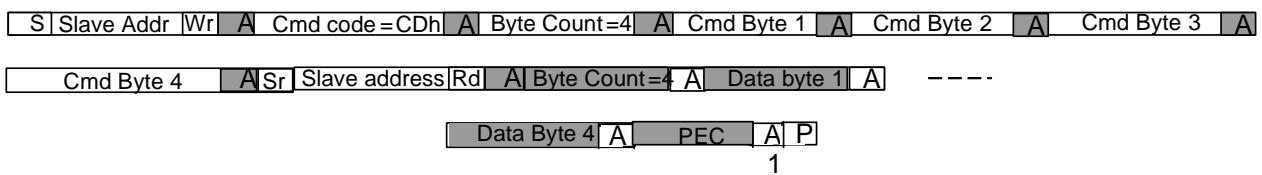
Using this command, the register to be read can be set up and read back with one SMBus cycle (a transaction with a START and ending in STOP). There is no STOP condition before the repeated START condition. The command format for the Block Write part of this command has the same sequence as in [Table 7-14](#), except that the Command Code changes to CDh, as illustrated below. Other Bytes remain the same as used in the sequence for SMBus Block Write followed by Block Read.

[Table 7-18](#) lists the Command format for Block Read.

**Figure 7-13. CSR Read Operation Using SMBus Block Read - Block Write Process Call**



**Figure 7-14. CSR Read Operation Using SMBus Block Read - Block Write Process Call with PEC**



**Table 7-18. Command Code for SMBus Block Read**

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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 7.3.4 SMBus Address Resolution Protocol

Address Resolution Protocol (ARP) is a protocol by which SMBus devices that implement an assignable Slave address feature are enumerated and dynamically assigned non-conflicting Slave addresses, rather than using a fixed Slave 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 7-15](#). 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 Slave address
- **Address Valid flag (AV)** – A flag bit or device internal state that indicates whether the device's Slave address is valid

The process of assigning a Slave 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 Slave 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 7.3.4.1 SMBus UDID

The 128-bit UDID is comprised of the following fields, as illustrated in [Figure 7-15](#) (not to scale). Each UDID field and its default value implemented in the PEX 8619 and meaning are explained in the tables that follow.

**Figure 7-15. 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 7-19. 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	Defaults to 10b. The PEX 8619 SMBus Address Type is implemented as Dynamic and volatile. 00b = Fixed address 01b = Dynamic and persistent 10b = Dynamic and volatile 11b = Random number device

**Table 7-20. SMBus Version/Revision [119:112]**

Field	Name	Default Value	Description
2:0	Silicon Revision ID	010b	PEX 8619, Silicon Revision BA.
5:3	UDID Version	001b	UDID version defined for <i>SMBus v2.0</i> .
7:6	<i>Reserved</i>	00b	

**Table 7-21. SMBus Vendor ID [111:96]**

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

**Table 7-22. SMBus Device ID [95:80]**

Field	Name	Default Value	Description
15:0	Device ID	8619h	PEX 8619 default Device ID value.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 7-23. SMBus Interface [79:64]**

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

**Table 7-24. SMBus Subsystem Vendor ID [63:48]**

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

**Table 7-25. SMBus Subsystem Device ID [47:32]**

Field	Name	Default Value	Description
15:0	Subsystem Device ID	8619h	PLX part number for the PEX 8619.

**Table 7-26. SMBus Vendor-Specific ID [31:0]**

Field	Name	Default Value	Description
31:0	Vendor-Specific ID	Depends upon I2C_ADDR[1:0] input levels. The four combinations provide the following ID values: 00b = 7000_0000h 01b = B000_0000h 10b = D000_0000h 11b = E000_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 combination of two Address bits produces four unique Vendor-Specific ID values, for a maximum of four SMBus-enabled PEX 8619s to co-exist on the same SMBus segment.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

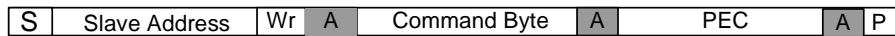
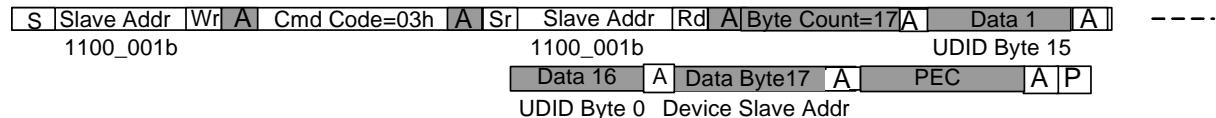
### 7.3.4.2 SMBus Supported ARP Commands

The PEX 8619 supports all ARP Slave commands. The Notify ARP Master command, which requires Master functionality, is *not* supported. [Table 7-27](#) explains the PEX 8619 response to each received ARP command.

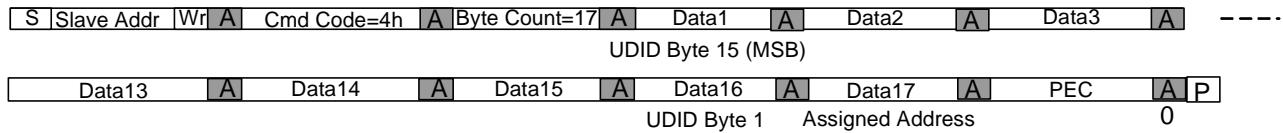
**Table 7-27. SMBus Supported ARP Commands, Format, and Actions**

ARP Command	SMBus Command Format	Slave Address	Command Code	Action
Prepare to ARP (Only General)	Send Byte (Refer to <a href="#">Figure 7-16</a> )	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 <a href="#">Figure 7-16</a> )	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 <a href="#">Figure 7-16</a> )	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 <a href="#">Figure 7-17</a> )	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 Slave address.
Assign Address ARP	Block Write (Refer to <a href="#">Figure 7-18</a> )	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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 7-16. Prepare SMBus ARP Command and SMBus Reset Device Command Format****Figure 7-17. Get SMBus UDID Command Format (General Get UDID Command with PEC)**

**Note:** If the **SMBus Configuration** register **AR Flag** bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[11]) is Cleared, the device returns the Slave Address field as 1111\_111b; otherwise, it returns the device Slave address. Bit 0 (LSB) in the Data Byte 17 field should be 1.

**Figure 7-18. Assign SMBus Address ARP Command Format**

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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 7.3.5 SMBus PEC Handling

The PEX 8619 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 8619 supports PEC Cyclic Redundancy Check (CRC) generation and checking during ARP, as well as during Read/Write transfers to the PEX 8619 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 8619 checks the PEC, if the Master transmits the additional PEC byte and the PEX 8619 PEC checking feature is enabled (default). PEC checking can be disabled, by Setting the **SMBus Configuration** register *PEC Check Disable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[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 8619 accepts the packet. During a register Read, if the Master requests the additional PEC byte, the PEX 8619 generates and transmits the PEC byte after the register data.

### 7.3.6 Addressing PEX 8619 SMBus Slave

By default, the PEX 8619 supports ARP when the I2C\_ADDR2 input is tied Low, and expects the ARP Master to Set the PEX 8619 SMBus Device Address. If ARP is disabled by I2C\_ADDR2 input being pulled High, the default Slave address is 38h (Address bits [7:1] are 0111\_000b, with Address bit [1:0] values loaded from the I2C\_ADDR[1:0] inputs). The two Address bits allow a maximum of four PEX 8619 SMBus Slaves to co-exist without address conflict on the SMBus, using SMBus Address byte values of 70h, 72h, 74h, and 78h. The **I2C\_ADDR[2:0]** inputs are loaded immediately after Fundamental Reset, and any subsequent change of input value does not affect functionality.

If the **SMBus Configuration** register *UDID Address Type* field is programmed as Fixed Address (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[13:12], are both Cleared) without disabling ARP, the PEX 8619 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 344h[7:1]). ARP can also be enabled or disabled at runtime, by writing to the register's *ARP Disable* bit (offset 344h[8]). 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 Slave 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 344h[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 344h[15]). Writes to this register bit take effect only when the register's **SMBus Command In-Progress** bit (offset 344h[28]) is Cleared, which indicates that the PEX 8619 SMBus interface is in the Idle state.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 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 8619, as a Slave-Transmitter, detects that it has pulled the I2C\_SCL0 line Low for more than 25 ms, the PEX 8619 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 8619 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 8619'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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 344h[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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 8 Virtual Channels and Port Arbitration

### 8.1

## Quality of Service Support

Quality of Service (QoS) is a feature offered by PCI Express that was not available before in PCI. *QoS* refers to being able to guarantee preferential treatment for a particular application or for traffic coming from a particular source. The PEX 8619 provides two methods for acquiring QoS:

- Virtual Channel support
- Port arbitration

### 8.2

## Virtual Channel Support

The PEX 8619 supports up to two Virtual Channels (VCs) that act as two completely independent paths through the switch, as defined in the *PCI Express Base r2.0*. Although the two VCs of a Port share the same physical wire/Links, the PEX 8619 has separate queues and separate credit mechanisms for each VC. At the egress stage of each Port, the PEX 8619 has three arbitration methods for determining which TLP from which VC to send on the wire/Link.

The elements of these features are described in the following subsections.

#### 8.2.1

### Traffic Class to Virtual Channel Mapping

The Traffic Class to Virtual Channel (TC/VC) mapping is Port-specific and must be configured and enabled for each Port. The default configuration for a PEX 8619 Port maps all TC[7:0] bits to VC0, as defined in the Port's **VC0 Resource Control** register *TC/VC0 Map* bits (offset *15Ch[7:1, 0]*). Any of the TC IDs (except TC0) can be mapped to the second VC (*that is*, VC1), by removing them from the *TC/VC0 Map* bits and adding them to the Port's **VC1 Resource Control** register *TC/VC1 Map* field for TC[7:1] (offset *168h[7:1]*). If any TCs are mapped to VC1, in addition to performing all steps outlined in [Section 8.4](#) (to set up PCI Express credits for VC1), VC1 must be enabled (**VC1 Resource Control** register *VC1 Enable* bit (offset *168h[31]*) is Set).

*Note: The actual VC ID of the second VC is Set by the **VC1 Resource Control** register *VC1 ID* field (offset *168h[26:24]*).*

#### 8.2.2

### Ingress VC Arbitration

The PEX 8619 supports hardware-fixed Round-Robin arbitration on the ingress Ports. Round-Robin servicing is performed between the two VCs, per Port. If any ingress Ports are targeting a congested Egress queue (Egress queue depth is greater than its programmable threshold), that ingress Port removes itself from the arbitration pool.

#### 8.2.3

### Egress VC Arbitration

The PEX 8619 supports three methods of arbitration on the egress Ports:

- Fixed-Priority VC Arbitration
- Round-Robin VC Arbitration
- Weighted Round Robin Arbitration

Each method is described in the sections that follow.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 8.2.3.1 Fixed-Priority VC Arbitration

The *PCI Express Base r2.0* defines the default Hardware-Fixed Strict Priority scheme based upon the inherent priority of the VC IDs, where VC1 has higher priority than VC0. The VC IDs alone determine the sequence in which transactions are sent. For this Arbitration scheme, the **Port VC Capability 1** register *Low-Priority Extended VC Counter* field (offset **14Ch[6:4]**) must be Cleared.

### 8.2.3.2 Round-Robin VC Arbitration

If the **Port VC Capability 1** register *Low-Priority Extended VC Counter* field (offset **14Ch[6:4]**) is programmed to 001b (default), VC0 and VC1 share the same Low-Priority Virtual Channel group. As a result, the PEX 8619 alternates selecting VC0 and VC1 in a fair, Round-Robin manner.

### 8.2.3.3 Weighted Round Robin Arbitration

The PEX 8619 also supports a Weighted Round-Robin (WRR) Arbitration scheme between the two VCs, as defined by the upstream Port's **VC WRRx** registers (Upstream Port, offsets **530h** through **53Ch**). Each Port has an 8-way time-slice [7:0], whose Setting determines which of the VCs take that time slot. (Refer to Section 14.16.1, "Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h).")

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 8.3 Queue Handling

The PEX 8619 implements queues at the ingress and egress Ports. There is arbitration capability for both Ingress and Egress queues. By default, the PEX 8619 attempts to send every packet on the Ingress queue to the Egress queue, until the Egress buffer fills. That results in optimum use of the PEX 8619 on chip buffering; however, it might create a situation in which one high-priority Port takes too much latency.

The packets within each Port's Egress queue are serviced on a first-in, first-out basis within each VC and Type (without violating Conventional PCI Ordering rules). The amount of queuing in the egress side is programmable, per VC. The **Port Egress TLP Threshold** register (offset F10h) establishes a range (minimum and maximum) for the Egress queue depth, for each VC. By reducing the thresholds, the Egress queue size is constrained. A smaller Egress queue means that newly arriving packets from a high-priority source do not have to wait as long for any earlier arriving packets from another source in the same VC, because the quantity of earlier arriving packets is reduced.

Each Port's Egress queue size is programmable, using the **Port Egress TLP Threshold** register fields. The **Port Upper TLP Counter** field (offset F10h[26:16]) allow for a maximum queuing of TLPs in the egress Port. If that maximum is crossed, the destination backpressures any source attempting to write to the destination queue. After the programmed minimum (established by the **Port Lower TLP Counter** field (offset F10h[10:0])) is crossed, the backpressure releases and the source can once again forward TLPs to that destination.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 8.4 VC1 Ingress Credit

TLPs consume PCI Express credits (and RAM space) upon entering the PEX 8619. When the TLPs are sent out of the PEX 8619 and an ACK is received, these credits (and Buffer space) are released. When a TLP enters the PEX 8619 and the internal credit engine has free credits available, an UpdateFC DLLP is generated, to replenish the credits used by the Transmitter. If there are no credits available, the credit engine waits until an ACK is received from a sent TLP before generating an UpdateFC DLLP.

Refer to [Section 14.16.13, “Device-Specific Registers – Ingress Credit Handler Threshold \(Offsets A00h – B7Ch\),”](#) for the default PCI Express credits of various Ports, based upon their Link widths.

### 8.4.1 VC1

A key element of enabling and using VC1 on a Port/Link is providing for PCI Express credits associated with the VC1 queues.

*To enable, and program the credits for, VC1:*

1. The PEX 8619 VC1 Global Enable defaults to a value of 1 (VC1 support is disabled) at reset.

To enable the VC1 capability, Clear the **Miscellaneous Control** register **VC1 (2nd VC) Disable** bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **28Ch[0]**).

This enables the PEX 8619 to support a second VC, but does not actually enable the VC.

2. Program the VC1 Credit Threshold registers, starting at offset A0Ch. (Refer to [Section 14.16.13, “Device-Specific Registers – Ingress Credit Handler Threshold \(Offsets A00h – B7Ch\),”](#))
3. Set the **VC1 Resource Control** register **VC1 Enable** bit (offset **168h[31]**).

VC1 is now enabled.

#### 8.4.1.1 VC1 Threshold Registers

The default values of Credit fields in the various VC1 Credit Threshold registers (in the offset A0Ch through B7Ch range) are zero (0), which equates to infinite credits. Because the PEX 8619 cannot support infinite credits, except for Non-Posted Payloads, it is *very* important that the Threshold registers are re-programmed before enabling VC1. The minimum value is 1 Header credit and 1 Payload credit or 1 MPS Payload credits (The Posted and Completion Payload minimum is 1 MPS. The Non-Posted Payload credit minimum is 1). Depending upon the Port configuration and MPS, it is possible that even using the minimum VC1 credits, more credits will be advertised than are actually available in the PEX 8619. If this is the case, the Credit Threshold registers for VC0 will also need to be re-programmed, to a smaller value, to compensate for the credits assigned to VC1.

*Note: Enabling VC1 and programming the VC1 Credit Threshold registers can have a direct effect on the PEX 8619 and system performance. Because of this possible side effect and the possibility of advertising too many credits, contact PLX Technology Technical Support for assistance, as seen needed, when using VC1.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 8.4.1.2 Programming the VC1 Global Enable and Threshold Registers

The Global Enable (offset 28Ch[0]) and Threshold (offsets A0Ch to A2Ch) registers can be programmed in three ways, using:

- Serial EEPROM
- I<sup>2</sup>C
- CPU (software)

If the serial EEPROM is used to Set the Global Enable bit and the VC1 Threshold Registers, nothing further is needed. However, if I<sup>2</sup>C and/or the CPU is used, a Hot Reset to the PEX 8619's upstream Port must be generated after the PEX 8619 Credit Threshold registers are programmed. The Hot Reset is required, to allow the internal Credit Engine to account for the new credits assigned to VC1.

### 8.4.1.3 Usage Example

The simplest example is if only Non-Posted TLPs, or Posted TLPs with a small Payload length, are used. In the example that follows, each Port and type is given two Headers. If one Header is preferred, it is necessary to re-program the appropriate **INCH Threshold VC1** register *UpdateFC High-Priority Threshold for Header Credit* fields.

Example configuration:

- MPS = 128 bytes
- Upstream Port = Port 0
- Downstream Port 1 needs VC1

Registers to write:

1. Program offset A0Ch (Port 0, Posted) to 408h.
2. Program offset A10h (Port 0, Non-Posted) to 401h.
3. Program offset A14h (Port 0, Completion) to 408h.
4. Program offset A24h (Port 1, Posted) to 408h.
5. Program offset A28h (Port 1, Posted) to 401h.
6. Program offset A2Ch (Port 1, Completions) to 408h.
7. Clear offset 28Ch[0], to enable support for VC1.

Following a Fundamental or Hot Reset, a value of 1 is written to the Port 0 and Port 1 **VC1 Resource Control** register **VC1 Enable** bit (offset 168h[31]).

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 9 Performance Features

### 9.1 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
- Ingress Resources
- Latency
- Queuing Options
- Read Pacing
- Dual Cast

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

105

## 9.2 DLLP Policies

DLLP rates can vary from 0 to 2 or more DLLPs/TLP. The PEX 8619 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 8619 DLLP transmission rate.

Typically, TLPs have higher transmission priority on the wire than DLLPs. The PEX 8619, 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 8619 can be programmed to alter its default DLLP policies, to emphasize improved TLP throughput, faster acknowledgement, more credit, or simplest behavior. The PEX 8619 default policies are designed to achieve optimal performance for most applications. Programmable choices for a DLLP policy, however, allow for further optimization.

### 9.2.1 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 Acknowledge Control Packet (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 8619 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* or *PCI Express Base r2.0*. The values provided in [Table 9-1](#) define some of the possible default values, in symbol times.

**Note:** The values listed in [Table 9-1](#) have been adjusted to be different than the *PCI Express Base r2.0* values, to compensate for hardware-specific latencies.

**Table 9-1. Sample ACK Latency Timer Values**

Maximum Payload Size (Bytes)	x8 Gen 2 (Symbol Times)	x4 Gen 2 (Symbol Times)	x2 Gen 2 (Symbol Times)	x1 Gen 2 (Symbol Times)
128	118	124	89	288
256	158	169	134	466
512	137	205	170	610

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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 r2.0*)). 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 Control Shadow** register ***ACK TLP Counter Timeout*** field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **664h[10:9]**) value controls the 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 8619 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 **1F8h[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 8619 can store depends upon the type and size of TLPs received. (Refer to [Section 9.3](#).) The PEX 8619 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.2.2 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. Refer to [Section 9.3](#) for details regarding credit amounts.

If the PEX 8619 is not transmitting TLPs (*that is*, the Transmit Link is idle), and credit to replenish the credit used becomes available, the PEX 8619 immediately transmits an UpdateFC DLLP to the connected device. This is an opportunistic, low-priority UpdateFC DLLP.

However, if the PEX 8619 is busy transmitting TLPs to the connected device, the PEX 8619 does not transmit an UpdateFC DLLP until a programmed threshold is crossed. The PEX 8619 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 – for each Half-Station Port, located in the **Ingress Credit Handler (INCH) Threshold** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h through B7Ch).

The example of UpdateFC options (provided in [Table 9-2](#)) chart how, for the various options, an UpdateFC is triggered. This example is for a traffic stream of six back-to-back, 256-byte Posted TLPs, using a x4 Port, where the maximum Posted Header credit is 25 and the maximum Posted Payload credit is 128. A 256-byte Payload requires 16 credits (1 credit per 16 bytes). Therefore, each TLP in this case consumes 1 Header and 16 Payload credits.

Once a high-priority UpdateFC is triggered, if there are sufficient on-chip resources to do so, the running credit deficit is fully restored. For most non-congested applications, it is likely that ample chip resources will exist, to fully restore credit with every UpdateFC. However, if resources are running low, only a portion of the running credit is restored. If the threshold for transmitting an UpdateFC remains crossed, then, as more resources become available, a subsequent DLLP is transmitted until the deficit is satisfied.

Selecting the 100% policy results in a high-priority UpdateFC for every TLP received. By itself, this policy results in 1 DLLP/TLP, without factoring in the ACK policy. The 75% policy triggers 1 DLLP for every 2 TLPs for this traffic load, which results in 0.5 DLLP/TLP without the ACK. The 50% policy results in 0.25 DLLP/TLP, and the 25% policy results in 0.16 DLLP/TLP.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 9-2. Example UpdateFC Options**

TLP Received	Running Credit Header, Payload Consumed/Total	25% Remains Triggers when 6 Header or 32 Payload Credits Remain	50% Remains Triggers when 12 Header or 64 Payload Credits Remain	75% Remains Triggers when 18 Header or 96 Payload Credits Remain	Less than 100% Remains Update ASAP
TLP0	24/25, 112/128	–	–	–	UpdateFC
TLP1	23/25, 96/128	–	–	UpdateFC	UpdateFC
TLP2	22/25, 80/128	–	–	UpdateFC	UpdateFC
TLP3	21/25, 64/128	–	UpdateFC	UpdateFC	UpdateFC
TLP4	20/25, 48/128	–	UpdateFC	UpdateFC	UpdateFC
TLP5	19/25, 32/128	UpdateFC	UpdateFC	UpdateFC	UpdateFC

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 9.2.3 Unidirectional DLLP Policies

For unidirectional traffic, the PEX 8619 DLLP policies allow the most-frequent DLLPs, because DLLPs do not interfere with TLPs. (DLLPs flow in the opposite direction of TLPs.)

The PEX 8619 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 **1F8h[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 8619 can immediately transmit an UpdateFC after receiving only the TLP Header. By transmitting an UpdateFC earlier, the total credit advertised can be minimized. For large Payload sizes (*such as 1,024 and 2,048 bytes*), reserve PEX 8619 resources only as necessary. By programming fewer credits and having a fast UpdateFC policy, the system does not run out of credits and the PEX 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 9.3 Ingress Resources

*Note: This section of the data book introduces the concept of Half Stations (which is sometimes mentioned, but not often, in the other chapters within this data book). Its use is necessary here, however, to “illustrate” their function.*

The PEX 8619 manages ingress credit on an even- and odd-numbered-Port basis; therefore, there are separate Credit Handler engines for the Even and Odd Ports. An internal hardware processing partition of Even or Odd Ports is referred to as a *Half Station*. All even-numbered-Ports belong to Half Station 0. All odd-numbered-Ports belong to Half Station 1.

For each Half Station, there are two central resources of on-chip ingress credit RAM – Header and Payload. The total Half-Station Header RAM size is 256 Header credits. The total Half-Station Payload RAM size is 2,048 Payload credits. A Header credit reserves 4 DWords of RAM storage, regardless of whether the Header size is 3 or 4 DWords. By design, a PEX 8619 Half Station reserves 12 Header credits and 32 Payload credits out of the RAM totals, for special hardware-specific storage needs.

The **STRAP\_PORTCFG[3:0]** inputs configure the quantity of enabled Ports, and the Link width of each Port. At initialization, the PEX 8619 optimally assigns the credits, based upon the selected Port configuration.

Header RAM stores TLP Headers, meaning that every Header credit advertised reserves one Header RAM location, and every TLP received on the PEX 8619 uses one Header RAM location. After subtracting the reserved operational Half Station Header RAM locations, there remains a net total of 244 Header entries available.

Payload RAM stores TLP Payload. A Payload credit is in units of 16 bytes. Of the 2,016 user-configurable entries, 4 credits must be allocated for each Posted and Completion Header credit advertised. These credits are used internally, for linking Posted and Completion TLP Payload to their respective Header.

Every Port receives and transmits the following three traffic types (packets):

- Posted (P)
- Non-Posted (NP)
- Completions (Cpl)

Each Port can be assigned to one or two Virtual Channels (VC0 or VC1).

All three traffic types and each Virtual Channel (VC0 and/or VC1) needs credit, for each Port. The **INCH Threshold** register bit fields (refer to [Table 9-6](#)) allocate the credit to be reserved (and advertised) for each Port VC, and traffic type. Once allocated, the credits remain dedicated to each Port, VC, and traffic type.

As TLPs arrive, they are stored on the PEX 8619, until an ACK is received from the final destination. Until the ACK is received, each TLP stored consumes credit, and continues to occupy RAM until released. The RAM/credit is released only after the Receiver has acknowledged to the sender the arrival of the TLP, without any errors per ACK/Negative Acknowledge (NAK) policy. The total quantity of TLPs stored, but not yet forwarded to, and acknowledged by, the next PCI Express device, depends upon congestion and the next PCI Express device’s ACK policy.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

There are trade-offs between the quantity of credits allocated for a particular traffic type and Port combination, perhaps more for one system configuration than another. To alleviate these trade-offs, the PEX 8619 contains an innovative Dynamic Buffering design that allows a programmable-sized portion of the RAM to store any of the three traffic-type TLPs from any Half-Station Port. The credits that remain, after allocating credits for each of the three traffic types and VC for each Port, become part of a *Dynamic buffer*. The Dynamic buffer is essentially a common pool of Half-Station credits, and is discussed in detail in [Section 9.3.2](#) and [Section 9.3.3](#).

The PEX 8619 default credit allocation values, which create a variable-sized Dynamic buffer for each of the possible Port configurations, are optimal for most applications. Detailed tables of the default initial credit allocation for all three TLP types, and an explanation of the common credit pool, are addressed in the following sections.

### 9.3.1 Initial Credit Allocation

The PEX 8619 default credit allocation values depend upon the strapped Link width, not the negotiated Link width. The initial credit values that the Initialization Flow Control (InitFC) DLLP advertises on a per-Port basis, which the PEX 8619 transmits after Linkup, are listed in [Table 9-3](#) (in credits) and [Table 9-4](#) (in bytes).

The amount of credit that a Port initially advertises is controlled by the **INCH Threshold** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h through B7Ch). (Refer to [Section 9.3.3](#).) The default value of these registers changes, depending upon the **STRAP\_PORTCFG[3:0]** input levels. Because of the way the hardware links the default **INCH Threshold** register credit values into actual reserved RAM storage, in some cases, one or two additional initial Payload credits are allocated out of the Half-Station common pool. If this occurs, the Port advertises this additional initial credit in its InitFC DLLPs.

**Note:** *The default initial credits for VC1 are all infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed into the **INCH Threshold** register fields, using a serial EEPROM.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 9-3. Initial Port Credit Allocation (where 1 Header credit means storage is available for one Header of any size, and 1 Payload Credit = 16 bytes)**

Configured Port Link Width	Posted Header/Payload	Non-Posted Header/Payload	Completion Header/Payload
<b>VC0</b>			
x8	49/384	49/infinite	49/320
x4	26/256	26/infinite	26/224
x1	7/64	7/infinite	5/64
<b>VC1</b>			
x8	infinite/infinite	infinite/infinite	infinite/infinite
x4	infinite/infinite	infinite/infinite	infinite/infinite
x1	infinite/infinite	infinite/infinite	infinite/infinite

**Table 9-4. Initial Port Credit Allocation for Header/Payload (where 1 Header credit means that storage is available for one Header, in bytes; Payload credits are listed in bytes)**

Configured Port Link Width	Posted Header/Payload	Non-Posted Header/Payload	Completion Header/Payload
<b>VC0</b>			
x8	784/384	784/infinite	784/320
x4	416/4,096	416/infinite	416/3,584
x1	112/1,024	112/infinite	80/64
<b>VC1</b>			
x8	infinite/infinite	infinite/infinite	infinite/infinite
x4	infinite/infinite	infinite/infinite	infinite/infinite
x1	infinite/infinite	infinite/infinite	infinite/infinite

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 9.3.2 Dynamic Buffering

The PEX 8619 default credit values are optimal for most applications, to maintain back-to-back TLP traffic indefinitely, without running out of credit. After any of the initial credit (storage space) is used, more resources are automatically made available from the Half-Station Dynamic buffer, to maintain the initial credit allotment. These additional resources, taken from the Dynamic buffer, are not pre-reserved. Therefore, they can be used for either Virtual Channel or any of the three TLP types – Posted, Non-Posted, or Completion (P, NP, or Cpl, respectively). Because the TLP type is not pre-specified, these extra resources are termed a *common credit pool*.

When credit is actually replenished depends upon the UpdateFC DLLP policy (refer to [Section 9.2.2](#)), which is controlled by Setting the **INCH Threshold** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h through B7Ch[19:18, 17:16], as appropriate). These thresholds are relative to the initial credits allocated to a Port. Common Pool credits that are allocated to a Port. Common Pool credits that are allocated to a Port before a high-priority UpdateFC DLLP is sent can be de-allocated, if the Port's initial credits are restored before the UpdateFC DLLP is sent.

The common credit pool for Header and Payload credit is as follows:

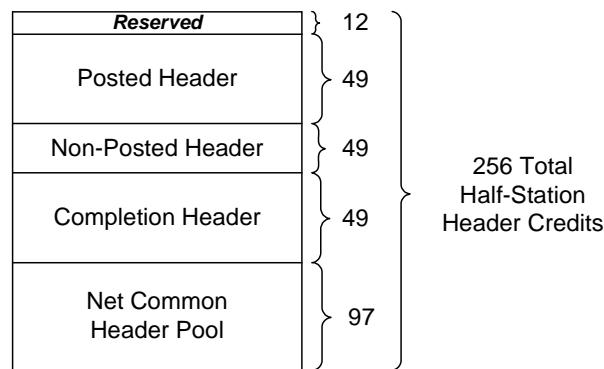
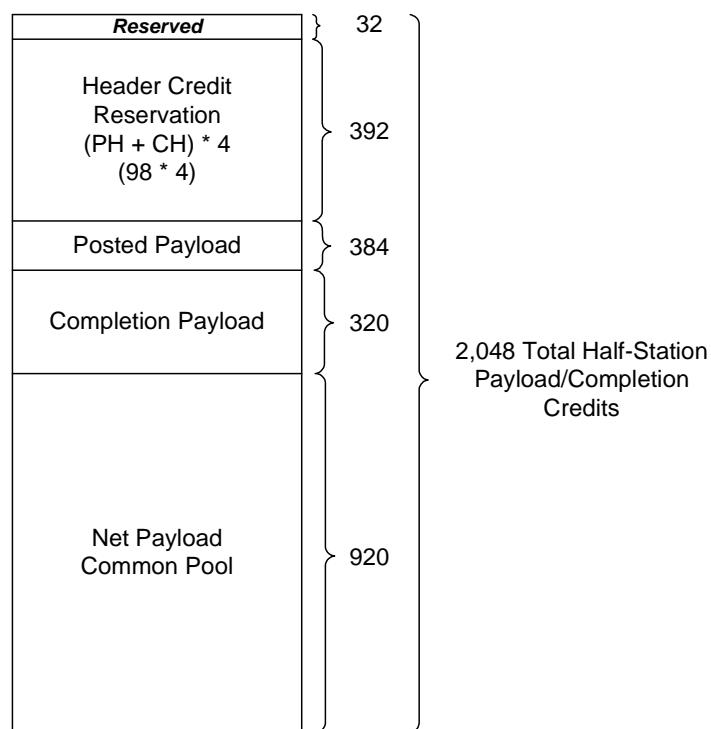
- **Common Header pool** – What remains in the Header RAM space, after subtracting the advertised Header credits of each Half-Station Port for each of the three TLP types, plus one or two VCs. [Figure 9-1](#) illustrates the way in which the initial allocation of the PEX 8619 Header RAM is partitioned (by default) for a Half-Station x8 Port, with only VC0 in use.
- **Common Payload pool (Common Payload/Completion pool)** – What remains after subtracting the following from the Payload RAM:
  - 4 credits for each Posted and Completion Header Credit advertised, for each Half-Station Port
  - Posted (Payload) credits advertised for each VC and Half-Station Port
  - Completion credits advertised for each VC and Half-Station Port

[Figure 9-2](#) illustrates the way in which the PEX 8619 Payload (and Completion) RAM is partitioned (by default) for one Half-Station x8 Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Figure 9-1. One x8 Port Default Configuration of Header RAM (256 Total Credits)****Figure 9-2. One x8 Port Payload/Completion RAM Default Configuration of 2,048 Credits**

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

A larger common pool allows the most flexibility for handling any possible instantaneous traffic stream, without backpressuring ingress flows. The PEX 8619's initial credit allocation default Settings leave sufficient on-chip RAM to accommodate numerous large TLPs in the common pool, after default values for the initial credits are subtracted. [Table 9-5](#) summarizes the common pool default allotment for several Half-Station Port configurations.

**Table 9-5. Half-Station Port Configuration Common Pool Default Allotments**

Half-Station Port Configuration	Common Pool Header Credits	Common Pool Payload Credits
x8	97	920
x4x4	88	640
x4	166	1,328
x4x1x1x1x1	90	624
Eight x1	92	608
x4x1	147	1,152
x1x1x1x1	168	1,312

*Notes:*

1. Actual RAM usage for TLP and Header storage and linking is variable.
2. The subtraction of 4 credits for each Posted and Completion Header credit advertised, for each Port, is a simplification – the Common Pool credit values can be larger.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.3.3 Ingress Credit Handler Threshold Registers (Offsets A00h through B7Ch)

*Note: Although the entire end range of the INCH Threshold registers is included in the register reference, only those offsets associated with active Ports are used. All others are either reserved or Factory Test Only, as identified in Section 14.16.13, “Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch).”*

For each Port, there are six sets of **INCH Threshold** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h through B7Ch) – Posted, Non-Posted, and Completion – one each, per VC. [Table 9-6](#) lists the lower 16 bits for each register. (To view the complete register set, refer to [Section 14.16.13, “Device-Specific Registers – Ingress Credit Handler Threshold \(Offsets A00h – B7Ch\).”](#))

The Non-Posted credits for Payload are Cleared, which equates to infinite credits. Because Non-Posted TLPs only have a 1-DWord Payload, they will never be longer than 5 DWords. Because the Header RAM is 5 DWords wide, only one Non-Posted Header credit is necessary to store a Non-Posted TLP.

**Table 9-6. INCH Threshold Registers (Offsets A00h through B7Ch), Payload and Header Credit Fields**

Ports and VC	Register Offset	Type	Payload		Header	
			Bit(s)	Description	Bit(s)	Description
Port 0, VC0	A00h	Posted	8:3	Payload Credit	15:9	Header Credit
	A04h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A08h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 0, VC1	A0Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	A10h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A14h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 1, VC0	A18h	Posted	8:3	Payload Credit	15:9	Header Credit
	A1Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A20h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 1, VC1	A24h	Posted	8:3	Payload Credit	15:9	Header Credit
	A28h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A2Ch	Completion	8:3	Payload Credit	15:9	Header Credit
Port 2, VC0	A30h	Posted	8:3	Payload Credit	15:9	Header Credit
	A34h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A38h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 2, VC1	A3Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	A40h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A44h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 3, VC0	A48h	Posted	8:3	Payload Credit	15:9	Header Credit
	A4Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A50h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 3, VC1	A54h	Posted	8:3	Payload Credit	15:9	Header Credit
	A58h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A5Ch	Completion	8:3	Payload Credit	15:9	Header Credit
Port 4, VC0	A60h	Posted	8:3	Payload Credit	15:9	Header Credit
	A64h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A68h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 4, VC1	A6Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	A70h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A74h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 5 VC0	A78h	Posted	8:3	Payload Credit	15:9	Header Credit
	A7Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A80h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 5 VC1	A84h	Posted	8:3	Payload Credit	15:9	Header Credit
	A88h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A8Ch	Completion	8:3	Payload Credit	15:9	Header Credit

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 9-6. INCH Threshold Registers (Offsets A00h through B7Ch), Payload and Header Credit Fields**

Ports and VC	Register Offset	Type	Payload		Header	
			Bit(s)	Description	Bit(s)	Description
Port 6 VC0	A90h	Posted	8:3	Payload Credit	15:9	Header Credit
	A94h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A98h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 6 VC1	A9Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	AA0h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AA4h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 7 VC0	AA8h	Posted	8:3	Payload Credit	15:9	Header Credit
	AACh	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AB0h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 7 VC1	AB4h	Posted	8:3	Payload Credit	15:9	Header Credit
	AB8h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	ABCCh	Completion	8:3	Payload Credit	15:9	Header Credit
Port 8 VC0	AC0h	Posted	8:3	Payload Credit	15:9	Header Credit
	AC4h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AC8h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 8 VC1	ACCh	Posted	8:3	Payload Credit	15:9	Header Credit
	AD0h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AD4h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 9 VC0	AD8h	Posted	8:3	Payload Credit	15:9	Header Credit
	ADCCh	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AE0h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 9 VC1	AE4h	Posted	8:3	Payload Credit	15:9	Header Credit
	AE8h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AECCh	Completion	8:3	Payload Credit	15:9	Header Credit
Port 10 VC0	AF0h	Posted	8:3	Payload Credit	15:9	Header Credit
	AF4h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AF8h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 10 VC1	AFCh	Posted	8:3	Payload Credit	15:9	Header Credit
	B00h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B04h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 11 VC0	B08h	Posted	8:3	Payload Credit	15:9	Header Credit
	B0Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B10h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 11 VC1	B14h	Posted	8:3	Payload Credit	15:9	Header Credit
	B18h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B1Ch	Completion	8:3	Payload Credit	15:9	Header Credit

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 9-6. INCH Threshold Registers (Offsets A00h through B7Ch), Payload and Header Credit Fields**

Ports and VC	Register Offset	Type	Payload		Header	
			Bit(s)	Description	Bit(s)	Description
Port 12 VC0	B20h	Posted	8:3	Payload Credit	15:9	Header Credit
	B24h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B28h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 12 VC1	B2Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	B30h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B34h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 13 VC0	B38h	Posted	8:3	Payload Credit	15:9	Header Credit
	B3Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B40h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 13 VC1	B44h	Posted	8:3	Payload Credit	15:9	Header Credit
	B48h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B4Ch	Completion	8:3	Payload Credit	15:9	Header Credit
Port 14 VC0	B50h	Posted	8:3	Payload Credit	15:9	Header Credit
	B54h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B58h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 14 VC1	B5Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	B60h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B64h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 15 VC0	B68h	Posted	8:3	Payload Credit	15:9	Header Credit
	B6Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B70h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 15 VC1	B74h	Posted	8:3	Payload Credit	15:9	Header Credit
	B78h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	B7Ch	Completion	8:3	Payload Credit	15:9	Header Credit

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.3.4 Adjusting Initial Credit Values (Ingress Resources)

The default Initial Credit values listed in [Table 9-3](#) and [Table 9-4](#) can be changed; however, to do so, the values must be changed by serial EEPROM and/or I<sup>2</sup>C, before the initial advertisement. It is also possible to use software to program the Credit registers over the Link; however, if the Link is up, credit cannot be removed, and values can only be increased. The Credit registers are sticky – a Hot Reset preserves any programmed values, and thereby allows any of the available programming methods to program credits at any time, even after the Link is up, if a Hot Reset is issued afterward to re-run the InitFC sequence.

When changing any credit value, follow the rules outlined in this section; otherwise, the credit can be incorrectly issued and data can be lost.

Credit is partitioned/programmed on a Half-Station basis. All even-numbered-Ports belong to Half Station 0. All odd-numbered-Ports belong to Half Station 1. Per the *PCI Express Base r2.0*, the minimum initial credit must be sufficient to meet the credit requirements of the MPS. To meet this requirement with a 2,048-byte MPS, the minimum credit value assigned to a Port, for both Posted and Completion TLPs, must be 128 credits each (one credit represents 16 bytes of storage). The MPS is restricted to 512 bytes when the programmed Link width is x1.

Additionally, because each TLP may not optimally fill each location in the internal RAM, the Header credit affects the Payload credit used to store the Payloads. Therefore, for every Posted or Completion Header credit reserved, 4 credits from the Payload/Completion RAM must also be held in reserve.

The following abbreviations are used in the rules outlined in this section:

- *PH* is the total Posted Header credits that are advertised or can be stored
- *NPH* is the total Non-Posted Header credits that are advertised or can be stored
- *CH* is the total Completion Header credits that are advertised or can be stored
- *MPS* is the Maximum Payload Size
- *Hmax* is the maximum quantity of Header credits that can be assigned, per Half Station
- *Pmax* is the maximum quantity of Payload and Completion credits

The total credit advertised, per Half Station, must follow these rules:

1. Sum of all Header credits  $\leq$  Hmax = 244.  
Sum of all Header credits = sum (all Ports PH + NPH + CH).
2. Payload and Completion credit must be sufficient for 1 MPS, for each Port.

*Note: Non-Posted Payload credit is infinite and Read-Only.*

3. Sum of all Payload and Completion credits assigned, per Half Station,  
is  $\leq$  Pmax = 2,016  $\leq$  (Advertised Posted Payload + Advertised Completion Payload  
+ (4 x (sum of all Ports PH + CH))).

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Programming Example** – To satisfy these rules, the following example is presented, using the default values of a Half Station strapped to have two x4 Ports, for VC0.

1. From [Table 9-3](#), the sum of all VC0 Header credits =  $2 \times (26 + 26 + 26) = 156$ , which is  $\leq H_{max} = 244$ .
2. For a 2,048-byte MPS, 128 credits must be allocated/advertised for both a Posted and Completion Payload, for each Port. Therefore, for the two Ports in the example, this uses  $2 \times (256 + 224) = 960$  credits, or 1,000 Payload/Completion credits.
3. For this case, the Payload/Completion credits used = 1,000 credits + 4 credits ( $2 \text{ Ports} \times (26 + 26)/\text{Port} = 1,416$ ) credits which is  $\leq P_{max} (2,016)$ .

For this example:

- Common Header Pool =  $(244 - 156) = 88$  Header Credits
- Common Payload/Completion Pool =  $(2,016 - 1,416) = 600$  Payload/Completion Credits

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.3.5

## Credit Allocation When Common Pool Is Consumed

A Half Station's common credit pool is consumed by any Port's Ingress or Egress queue for that Half Station, on a first-come, first-served basis. If the Half Station's common pool is completely consumed (and therefore, each Half-Station Port's credit), the PEX 8619 Half Station is in a congested state. In the congested state, the Half Station is unable to provide additional credit to any of its Ports, until credit is released. Credit is released only after the Receiver acknowledges the packet with an ACK. In this state, the following register fields provide a method for users to weight each Port's Request for more credit, from the Half Station's internal credit allocation logic:

- **INCH Threshold Port x VC0 Posted** register *Congested Port Weight* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h, A18h, A30h, A48h, A60h, A78h, A90h, AA8h, AC0h, AD8h, AF0h, B08h, B20h, B38h, B50h, B68h[22:20])
- **INCH Threshold Port x VC1 Posted** register *Congested Port Weight* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A0Ch, A24h, A3Ch, A54h, A6Ch, A84h, A9Ch, AB4h, ACCh, AE4h, AFCh, B14h, B2Ch, B44h, B5Ch, B74h[22:20])

In the congested state, the Half Station's internal credit allocation logic decides which Port will receive the next available credit, by evaluating the following:

- Each VC and Port's *Congested Port Weight* field Setting
- Quantity of Common Pool credits that the Port has already consumed
- History of which Ports have recently received credit

**Table 9-7**, By default, the *Congested Port Weight* field for each VC and Port is Cleared. The default value is called an *effective rate Setting*. For the default case, if a Half Station is configured into five Ports (x4, x1, x1, x1, x1), each Port receives credit updates based upon the Port's negotiated Link width. The x4 Port would receive 50% of the credit updates, and the four x1 Ports would each receive effectively 1/8<sup>th</sup> of the remaining credits as they become available.

**Table 9-7** defines the *Congested Port Weight* field values. Requests are weighted, based upon the Port's effective Link width, relative to the effective Link widths of the other Half Station's Ports. Settings can reduce or increase a Port's effective rate, down to x1 or up to x8, respectively. (*That is*, regardless of the actual Link width, a Link can only be reduced to a x1 effective rate, or increased to a x8 effective rate.) The effective Link Width Request weight is calculated, by multiplying the Port's negotiated Link width (not strapped width) by the *Congested Port Weight* field Setting.

The Half Station's internal credit allocation logic decides how to allocate the Common Pool credits, as they become available in a congested scenario. Regardless of the Common Pool credit availability, each Port maintains ownership of the ingress credits that were initially allocated to it. In a congested state, those dedicated credits are replenished to their assigned Port, after an ACK is received from the final destination.

The effective rate Setting applies to both the Common Header and Common Payload/Completion pools, for the selected Half Station, Port, and VC. Although the effective rate Setting is in the **INCH Threshold Port x VC0 Posted** and **INCH Threshold Port x VC1 Posted** registers, the value applies to credit updates for all three possible traffic types (Posted, Non-Posted, and Completion).

*Note: The offsets for these registers are listed on the previous page.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 9-7. INCH Threshold Port x VC0 Posted and INCH Threshold Port x VC1 Posted Register *Congested Port Weight* fields  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, field [22:20]) Values**

<b>Congested Port Weight Setting (Bits [22:20])</b>	<b>Description</b>
000b = eff_rate	Request is weighted, based upon the Port's Link width relative to the effective Link widths of the other Half Station's Ports.
001b = 2x eff_rate	Increases the weight of a Request by 2 ( <i>for example</i> , a x4 Link Width Request weight increases to a x8 Link Width Request weight).
010b = 4x eff_rate	Increases the weight of a Request by 4 ( <i>for example</i> , a x1 Link Width Request weight increases to a x4 Link Width Request weight).
011b = 8x eff_rate	Increases the weight of a Request by 8 ( <i>that is</i> , a x1 Link Width Request weight increases to a x8 Link Width Request weight).
100b = 0	Port receives no credit from the common pool, until a decongested state is reached.
101b = eff_rate/2	Decreases the weight of a Request by 2 ( <i>for example</i> , a x8 Link Width Request weight decreases to a x4 Link Width Request weight).
110b = eff_rate/4	Decreases the weight of a Request by 4 ( <i>for example</i> , a x8 Link Width Request weight decreases to a x2 Link Width Request weight).
111b = eff_rate/8	Decreases the weight of a Request by 8 ( <i>that is</i> , a x8 Link Width Request weight decreases to a x1 Link Width Request weight).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.3.6 INCH Port Pool Registers (Offsets 940h through 94Ch)

The **INCH Port Pool** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) are registers whose original intent was to provide another level of reservation for Common Pool credits. The registers are named as follows, and are located at:

- **INCH Port Pool for Ports 0, 1, 2, 3**, offset **940h**
- **INCH Port Pool for Ports 4, 5, 6, 7**, offset **944h**
- **INCH Port Pool for Ports 8, 9, 10, 11**, offset **948h**
- **INCH Port Pool for Ports 12, 13, 14, 15**, offset **94Ch**

These registers are essentially redundant to what is accomplished by changing the values of the **INCH Threshold** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets A00h through B7Ch).

**Consider the INCH Port Pool registers to be reserved and only change the credit Settings, using the INCH Threshold registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.**

The **INCH Port Pool** registers initial values are provided in sets of two for each Port – Payload pool and Header pool. [Table 9-8](#) lists the bit decode for each Port.

The initial values of the **INCH Port Pool** registers are all Cleared which means that, by default, there is no additional level of reservation. Additionally, the **INCH Threshold** registers default values evenly allocate all available credit, across all enabled Ports.

**Table 9-8. INCH Port Pool Settings (Offsets 940h through 94Ch)**

Ports	Payload Pool Bit(s) <sup>a</sup>	Header Pool Bit(s) <sup>a</sup>
0, 4, 8, 12	2:0	6:4
1, 5, 9, 13	10:8	14:12
2, 6, 10, 14	18:16	22:20
3, 7, 11, 15	26:24	30:28

*a. The hyperlinks provided to the bits listed are to register offset 940h.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 9.3.7 Wait for ACK – Avoiding Congestion

Once a TLP arrives, it remains on the PEX 8619 until it is no longer required. The TLP can quickly egress the PEX 8619. However, until an ACK is received, indicating that the TLP was correctly received, each TLP must remain on the PEX 8619 and be ready to be re-sent multiple times. While on the PEX 8619, the TLP continues to use Half-Station common pool resources.

The *PCI Express Base r2.0* recommends sending an ACK within the approximate time it takes to send 1.5 to 3 MPS TLPs. It does not, however, suggest that smaller TLPs obtain faster ACKs. This data book describes the way in which the PEX 8619 sends an ACK. However, the PEX 8619, has no way of knowing its Link partner's ACK policy.

To minimize the amount of TLPs stored on the PEX 8619 while waiting for an ACK, follow these guidelines:

- Avoid traffic patterns where a great deal of back-to-back TLP bytes travel from a wide Link to a single narrow Link, because the narrow Link can only forward TLPs at a fraction of the ingress rate. *For example*, if a 4-KB MRd is transmitted upstream from a x4 Port and the upstream Port is x8, the upstream Port transmits a 4-KB CPLD to the Requester, two times faster than the Requester can receive the data. If the Requester transmits many of these MRd Requests, large amounts of CPLD data that require storage on the PEX 8619 will quickly accumulate.
- If there are many small TLPs, determine whether the PEX 8619's ACK response time can be reduced, as per the *PCI Express Base r2.0* guidelines.
- Evenly space the TLP pattern, rather than use a burst of many back-to-back TLPs followed by a long stall, to even the distribution and accommodate a fixed ACK Transmission Latency Timer.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 9.4 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 9-3 and Table 9-9 illustrate 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 8619 always cuts the packet through. The PEX 8619 has the same latency, regardless of whether the traffic is upstream or peer-to-peer.

As expected with the PEX 8619 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. This is indicated by the shaded-green entries in Table 9-9.

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:

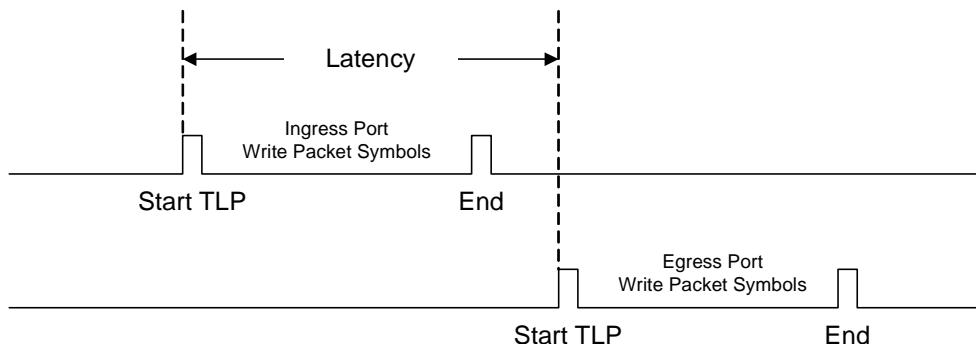
$$F = (E-I) / E$$

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

**Figure 9-3. Start TLP-to-Start TLP Latency Measurement**



**Table 9-9. Sample STP-to-STP Latency**

Latency for TLP with a Data Payload of 4/64/256 Bytes (in ns)				
	From Ingress			
	x8 Gen 2	x4 Gen 2	x2 Gen 2	x1 Gen 2
To Egress				
x8 Gen 2	142/138/138	148/166/220	162/222/378	182/310/678
x4 Gen 2	146/146/148	148/154/138	164/214/314	184/310/614
x2 Gen 2	146/146/146	150/150/166	174/180/180	200/296/488
x1 Gen 2	146/146/146	180/180/180	188/188/188	180/180/188

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.4.1 Host-Centric Latency

Host-centric traffic flows only to or from the Host. Host-centric latency depends upon the quantity of active streams. If there is only one active stream, or if the total Host bandwidth is greater than or equal to the sum of all traffic streams, the traffic is well-balanced and the latency measurements provided in [Table 9-9](#) apply.

If there is more traffic than an upstream Host can sink, congestion occurs when all the TLPs concurrently attempt to use the limited Host bandwidth. The latencies in that case depend upon the level of traffic congestion. In this case, Host bandwidth is at 100%, but the total downstream bandwidth is more than the Host bandwidth, and latencies continue to increase until the congestion eases.

Another case of increased latency is if the Host serially sends large amounts of Read Completion data to one downstream Port and then another downstream Port. *For example*, if the upstream Port is x8 and the two downstream Ports are both x4, it appears that there should not be a latency build-up, because the bandwidth matches. However, if the Host cannot interleave the destinations, one destination must wait until the Host completes transmitting traffic to the other destination, before it can receive any Read Completion data. In this case, the round-trip Read latency can significantly increase.

*For example*, suppose that one downstream Port transmits 16, 4-KB MRd Requests upstream. Those Read Requests represent 64 KB of data. If the upstream Port is x8 and the downstream Port is x4, the Read Completions back up into the PEX 8619, perhaps all the way to the Root Complex. Suppose another downstream Port transmits only one, 1-KB MRd Request upstream, shortly after the 16, 4-KB MRds were received by the Root Complex. For many Root Complexes, this one, 1-KB Read Request from the second device must wait for the 16, 4-KB MRd Requests from the first device to complete before being serviced. The PEX 8619 buffer is approximately 10 KB; therefore, the second downstream device must wait for (64-10) 54 KB of Completion data to transmit across a x4 Link before it begins to receive its Read Completions. On a x4 Link, 54 KB takes about 48  $\mu$ s, which significantly increases the second device's latency. The PEX 8619 contains Read Pacing logic that prevents this type of latency increase that occurs when multiple devices concurrently read data from the Root Complex. (Refer to [Section 9.6](#).)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 9.4.2 Peer-to-Peer Latency

If an Egress Port queue is not established (*such as* multiple streams to the same destination Port), peer-to-peer latencies match the best-case values listed in [Table 9-9](#). The PEX 8619 has the same latency, regardless of whether the traffic is Host-centric or peer-to-peer. Latency is constant in the non-congested case, no matter the source Port nor destination Port, if the Source Port has the same or greater bandwidth than the destination Port.

The discussion for Host-centric traffic applies to all Ports for peer-to-peer traffic. It is recommended that there be a method outside the scope of this data book, however, for balancing traffic flow for peer-to-peer applications.

## 9.4.3 Other Latency Measurements

In addition to STP-to-STP latency, there are other latencies to consider. [Table 9-10](#) lists various best-case latencies for several Link widths and speeds. Transmitted DLLPs can be required to wait for a TLP. DLLP policies can prevent sending a DLLP for a time period longer than the best case.

**Table 9-10. Miscellaneous Best Case Link Latencies (in ns)**

Latency <sup>a</sup>	x8 Gen 2	x4 Gen 2	x2 Gen 2	x1 Gen 2
STP in to UpdateFC SDP <sup>b</sup>	124	132	144	180
TLP's END in to ACK SDP	72	72	72	72
UpdateFC SDP in to STP	112	116	120	140

*a. Gen 1 latency values are expected to be the same as, or very close to, Gen 2 latency values.*

*b. SDP is “Start DLLP”.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

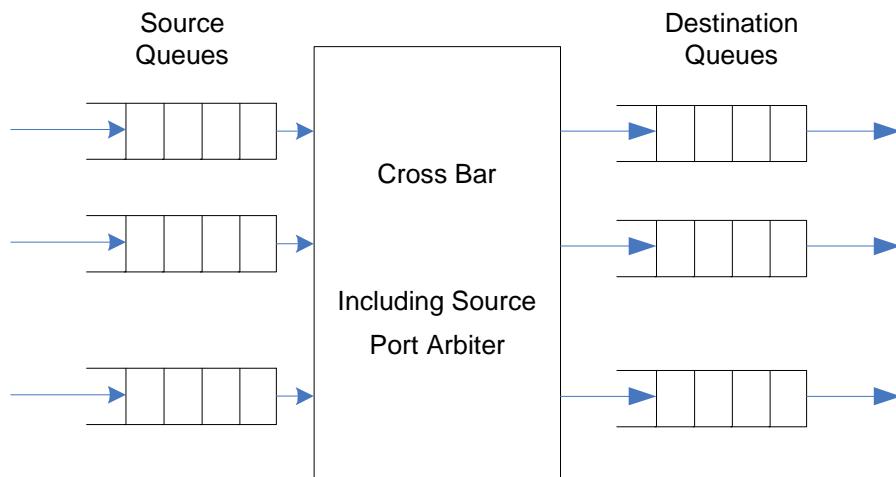
## 9.5 Queuing Options

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 8619. The PEX 8619 provides two alternatives, as to where to locate such queuing (refer to [Figure 9-4](#)):

- **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 9-4. On-Chip Queuing**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 9.5.1 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 8619 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.5.2 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 a few Ports, while the latency for other Ports is not as important.

Source queuing limits the Destination queue depth. When the Destination queue reaches the maximum depth, any subsequent TLPs targeting that Port are not forwarded, but are queued up in a per-Source Port-based queue. The Source Port queue does not forward TLPs until the Destination queue drops to a programmed threshold, upon which TLP forwarding is re-enabled.

**Note:** *A Source Port queue that cannot forward to a Destination queue blocks all subsequent TLPs arriving on that same Source Port, although the target Port is a different destination.*

The **Port Egress TLP Threshold** register (offset F10h) controls the minimum and maximum queue depths. **Table 9-11** 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 9.5.1](#), 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 9.5.3](#).

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 9-11. Port Egress TLP Threshold Register Port Lower and Upper TLP Counters (Offset F10h)**

Bit(s) <sup>a</sup>	Name	Description
10:0	<b>Port Lower TLP Counter</b>	When Source Scheduling is disabled due to the Port Upper TLP Counter (threshold) being exceeded, Source Scheduling is re-enabled when the Port TLP Counter goes below the Port Lower TLP Counter (this threshold). Because the default Setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8619, the Source Scheduler is disabled, by default.
26:16	<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. Because the default Setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8619, the Source Scheduler is disabled, by default.

a. Bits not identified in [Table 9-11](#) are reserved.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.5.3 Port Arbitration

In the crossbar that connects the Source queues to the Destination queues, there is a Port Arbiter for each Destination Port. The Port Arbiter ensures that each Source Port receives a deterministic bandwidth connecting to a Destination Port. The Port Arbiter ensures that each Source Port receives a deterministic bandwidth connecting to a Destination Port. Every Port has two default fixed Round-Robin Port Arbiters, one for each VC.

In addition to the default fixed Round-Robin Port Arbiter, there is one Device-Specific Weighted Round-Robin (WRR) Port arbitration hardware resource that can be enabled by system software. The Device-Specific WRR arbitration is also Round-Robin, but with programmable weighting for a particular Port or Ports. When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0.

System software discovers the Port Arbitration Capability, as reflected in the **VC0 Resource Capability** register (offset 158h[1:0]). If the system software needs to make use of an advertised WRR arbitration with 32-phase capability for a Port, it programs the Port's **VC0 Resource Control** register *Port Arbitration Select* field (offset 15Ch[19:17]) to 001b.

The WRR Source Port Arbiter has a 32-phase Port Arbitration Table, as outlined in the *PCI Express Base r2.0*, and documented in the **Port Arbitration Table Phase x** registers (When WRR is enabled, Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0, offsets 1A8h through 1B4h). (Refer to the *PCI Express Base r2.0*, as well as [Section 14.13.1, “WRR Port Arbitration Table Registers \(Offsets 1A8h – 1BCh\),”](#) for further details.)

Once one or more Phase registers are written, the software writes the Port's **VC0 Resource Control** register *Load Port Arbitration Table* bit (offset 15Ch[16]). When written, the register values are transferred to the WRR arbitration logic, and immediately take effect.

Port arbitration makes decisions on a per-TLP basis. A Port with more short TLPs will appear to receive less bandwidth, compared to a Port with fewer long TLPs, if both Ports have the same weight and both target a congested Port.

### 9.5.4 Port Bandwidth Allocation

For applications that need to allocate a fixed bandwidth to each Port, the PEX 8619 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.6 Read Pacing

**Caution:** *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 only on the upstream Port.

PCI Express has a weakness concerning the quantity of outstanding bytes requested by Reads. It is possible that a single device can overwhelm the system with a reasonable quantity 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 8619's Read Pacing capability:

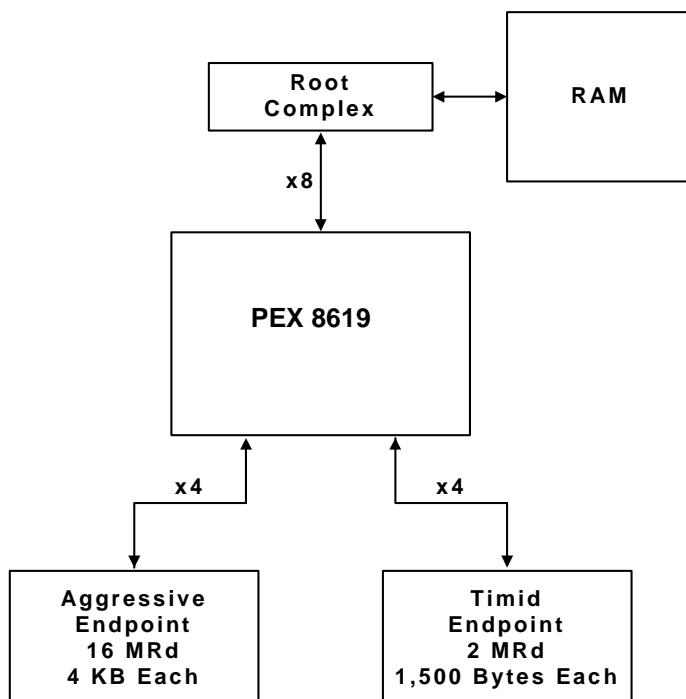
- Read spacing
- Read threshold

The following sections provide examples and further information regarding Read Pacing.

### 9.6.1 Read Pacing Example

Figure 9-5 illustrates an example of a system that benefits from Read Pacing.

**Figure 9-5. Read Pacing Example**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

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 9-5](#), 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>.

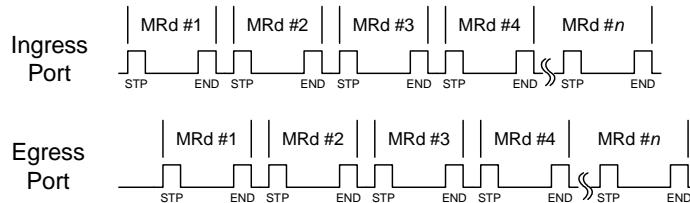
If either endpoint is running by itself, neither sees a problem. However, if both endpoints are concurrently active, the aggressive endpoint dominates the Root Complex Memory Controller. In addition, due to the bandwidth mismatch, Completions can queue up in the PEX 8619, creating too many Completions for the switch to store at one time. As a result, the PEX 8619 backpressures the Root Complex for Completions. The Root Complex can only forward Completions to the PEX 8619 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 quantity of Completions queued up in the PEX 8619 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 8619, not at the upstream Link's capacity.

[Figure 9-6](#) illustrates how a PCI Express switch, without Read Pacing, forwards Memory Read Requests (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 8619 functions when Read Pacing is enabled, and Read Spreading is enabled or disabled.

**Figure 9-6. 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.6.2 Read Spacing (Spreading) Logic

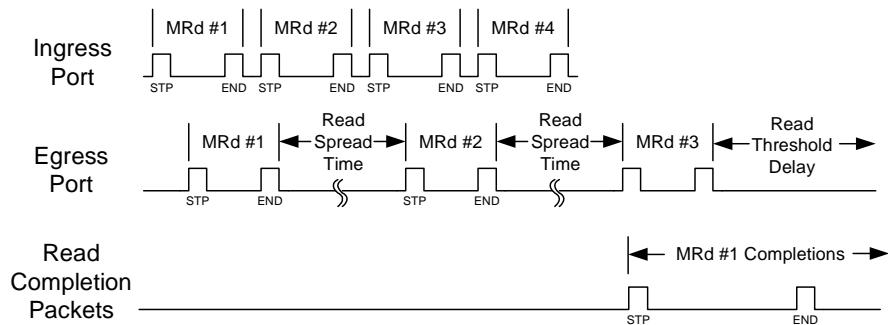
Read Spacing (also referred to as *Read Spreading*) spreads out Read Requests. The PEX 8619 Read Spacing logic looks at the Read Request size and the endpoint's bandwidth, to determine how often to forward subsequent Read Requests. *For example*, Read Requests arriving on a x1 Link can only sink data at a x1 rate. If a x1 endpoint submits multiple Read Requests to a x4 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 **Read Spreading Control** register *Port x Memory Read Spreading Disable* bits (offsets 544h[15:0] and 548h[15:0], respectively) must both be Cleared.

Figure 9-7 illustrates the way in which the PEX 8619 forwards Read Requests when the Read Pacing and Read Spreading-related bits are enabled. (Refer to Section 9.6.5 for additional register/bit information.) The PEX 8619 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 (Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port, offsets 54Ch and 550h, respectively).

**Figure 9-7. Read Pacing On (Enabled) and Read Spreading On (Enabled)**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

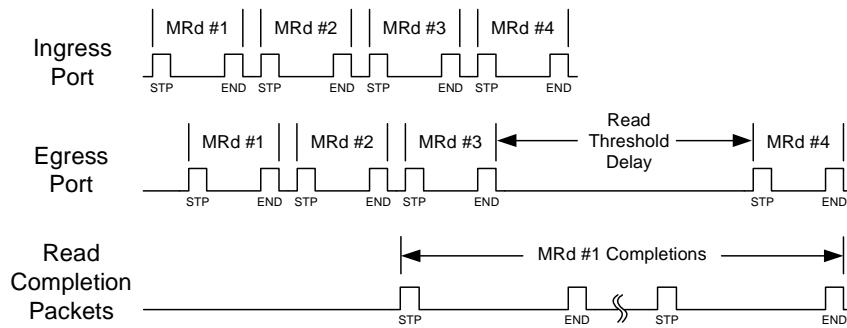
Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.6.3 Read Threshold

The Read threshold is the maximum quantity 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 8619'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 8619, waiting for Completion data to reduce the outstanding count to below the threshold. If an overabundant quantity of Read Requests queue in the PEX 8619, no additional Read credit is allocated, which backpressures the Read Requester. [Figure 9-8](#) illustrates the way in which the PEX 8619 forwards Read Requests when its Read Spacing logic is enabled and Read Spreading logic is disabled.

**Figure 9-8. Read Pacing On (Enabled) and Read Spreading Off (Disabled)**



### 9.6.4 Read Pacing Benefits

When Read Pacing logic is enabled, the PEX 8619 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 8619'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.

## 9.6.5 Enabling Read Pacing and Read Spreading

**Caution:** *Source Queuing and **Read Pacing** should not be concurrently enabled.  
The two features are incompatible and doing so can result in Fatal errors.*

Read Pacing is disabled, by default. To enable Read Pacing, the Port's **Read Pacing Control** register *Port x Read Pacing Disable* bit (Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port, offset **544h[15:0]**) must be Cleared. A bit value of 0 enables Read Pacing, whereas a value of 1 (default) disables Read Pacing.

The Port's **Read Spreading Control** register *Port x Memory Read Spreading Disable* bit (Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port, offset **548h[15:0]**) is used to enable or disable Read Spreading. A value of 1 disables Read Spreading for the corresponding Port. 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 9-12](#). (For complete details, refer to the register offset **54Ch** and **550h** description provided in [Section 14.16.2, "Device-Specific Registers – Read Pacing \(Offsets 544h – 554h\)"](#).) [Figure 9-6](#) through [Figure 9-8](#) 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 0** register controls the threshold values for x4 and x8 Link widths, and the **Read Pacing Threshold 1** register controls the threshold values for x1 Link widths (Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port, offsets **54Ch** and **550h**, respectively). The thresholds are in DWords. Narrower Link widths have lower thresholds, because they must buffer smaller quantities.

**Table 9-12. Read Pacing and Memory Read Spreading Control Register Disable Bits**

Register Offset	Bit(s) <sup>a</sup>	Description	Default
544h	15:0	<b>Port x Read Pacing Disable</b> Bits [15:0] correspond to the Read Pacing Disable for Ports 15 through 0, respectively. 0 = Read Pacing is enabled for this Port 1 = Read Pacing is disabled for this Port	FFFFh
548h	15:0	<b>Port x Memory Read Spreading Disable</b> Bits [15:0] correspond to the Memory Read Spreading Disable for Ports 15 through 0, respectively. 0 = Memory Read Spreading is enabled for this Port 1 = Memory Read Spreading is disabled for this Port	0000h

a. Bits not identified in [Table 9-12](#) are **Reserved or Factory Test Only**.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.7 Dual Cast

### 9.7.1 Introduction

This section describes the functions and programming of the PEX 8619's Dual Cast feature. A typical system configuration and register programming example is also provided.

Dual Cast allows programs to concurrently write the same data to two different destinations. Whenever Posted Memory Write TLPs entering the PEX 8619 through a designated Port (referred to as the *Dual Cast Source Port*) are addressed to designated memory regions (referred to as *Dual Cast BARs*), the switch automatically generates a copy of the original TLP (referred to as the *Dual Cast Copy TLP*), replacing the original TLP's address with one that is mapped to an egress Port designated as the *Dual Cast Destination Port*.

When an incoming TLP is copied in this manner, both the original TLP and Dual Cast copy TLP are concurrently queued at their respective egress Ports, effectively doubling the PEX 8619's egress rate for the same ingress rate.

### 9.7.2 Dual Cast System Model

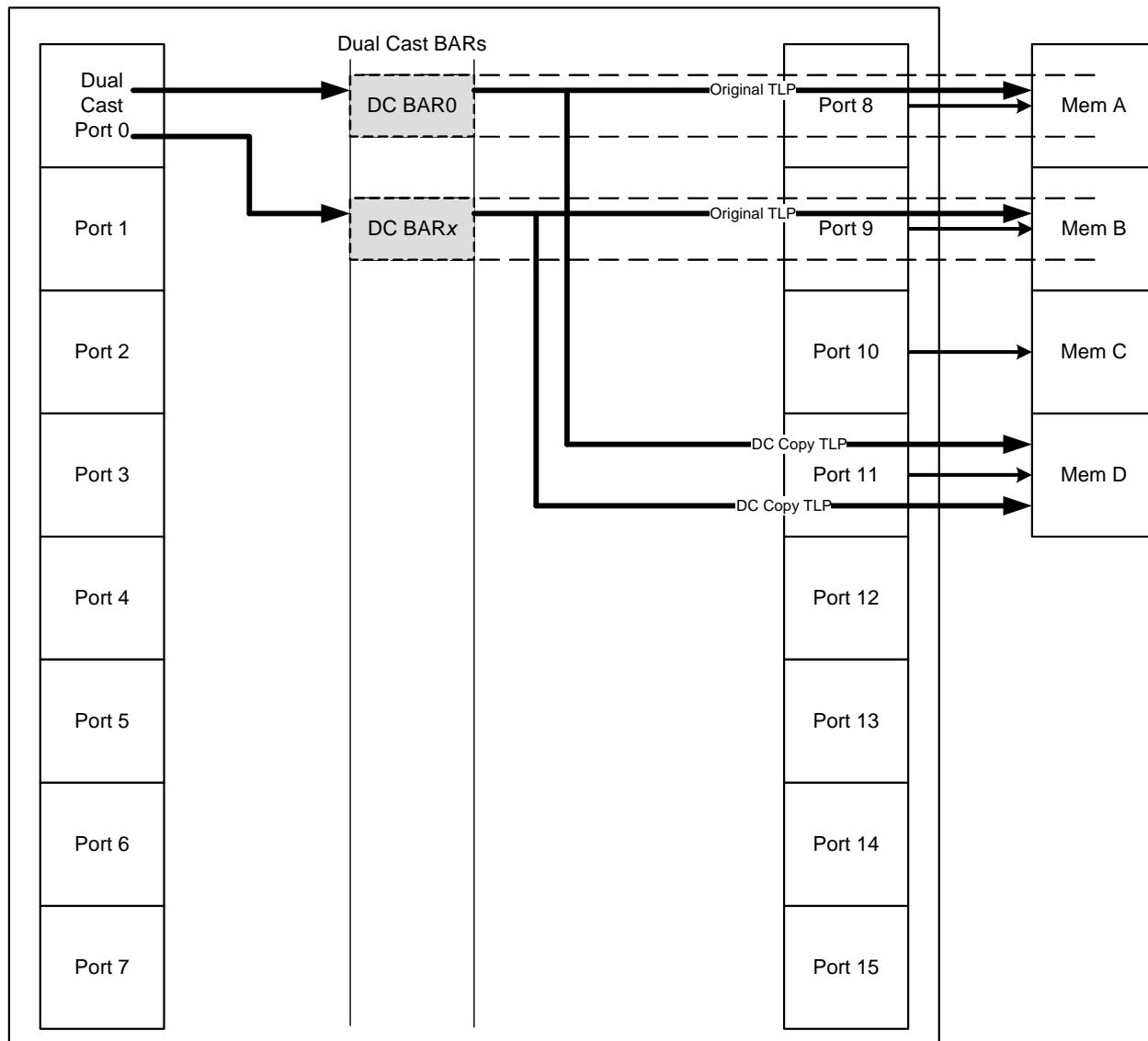
Figure 9-9 illustrates Dual Cast functions. In this figure, the designated Dual Cast Source Port is Port 0. Only Memory Write TLPs that enter the PEX 8619 through this Port (or alternately, any Port) are subject to Dual Casting. Dual Cast Control registers allow the programmer to define up to eight separate Dual Cast BARs, over which Dual Casting will be applied. If a Memory Write TLP enters the PEX 8619 through the designated Dual Cast Source Port, **and** its Header address falls within an active Dual Cast BAR, then the switch automatically generates a Dual Cast Copy TLP, replacing the original TLP's Header address with a new address mapped to the designated Dual Cast Destination Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

Dual Cast BARs can be mapped to any egress Port of the PEX 8619. In other words, original Memory Write TLPs that are being copied can exit the PEX 8619 on any egress Port, as they would normally. However, Dual Cast Copy TLPs, regardless of which Dual Cast BAR generates them, must all exit the PEX 8619 through the designated Dual Cast Destination Port.

The Dual Cast Destination Port can also be configured as NT. Refer to [Section 9.7.5](#) for additional programming requirements.

**Figure 9-9. Typical Dual-Cast System Model**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 9.7.3 Dual Cast Control Registers

This section describes each of the Dual Cast Control registers. Eight sets of identical registers describe each of the eight Dual Cast BARs. An additional register specifies the Dual Cast Source and Destination Ports. These registers are described, in detail, in the following subsections. For a complete listing of the registers, refer to [Section 14.14.10, “Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability \(Offsets 448h – 51Ch\).”](#) The Dual Cast registers are located in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port.

### 9.7.3.1 Dual Cast Low BAR[0-7], Dual Cast High BAR[0-7]

The 32-bit **Dual Cast Low** and **High BAR<sub>x</sub>** registers in Port 0 (refer to [Table 9-13](#)) are used to define the 64-bit physical Base address of each of the Dual Cast BAR address windows, numbered 0 through 7.

**Dual Cast Low BAR<sub>x</sub>** contains the lower 32-bits of the Base address of Dual Cast BAR window *n*. Only the upper 12 bits [31:20] of this register are used to specify address. Bits [19:0] of this register are not de-coded, and are hardwired with the value 0\_000Ch. Thus, Dual Cast BARs are naturally aligned on 1-MB boundaries. Because the 1-MB boundary is also a 4-KB boundary, any Memory Write TLP that falls within the Dual Cast BAR’s Address range will be guaranteed never to exceed the top of the Dual Cast BAR<sub>x</sub> range.

**Dual Cast High BAR<sub>x</sub>** contains the upper 32 bits of the Base address for Dual Cast BAR window *n*. For Base addresses in the lower 4 GB or for all 32-bit systems, this register should be Cleared (default).

**Table 9-13. Dual Cast High/Low BAR<sub>x</sub> Register Locations**

Base Address Register	Port 0 Register Offset	
	Dual Cast Low[31:0] (Lower 32 Bits)	Dual Cast High[31:0] (Upper 32 Bits)
BAR0	450h	454h
BAR1	468h	46Ch
BAR2	480h	484h
BAR3	498h	49Ch
BAR4	4B0h	4B4h
BAR5	4C8h	4CCh
BAR6	4E0h	4E4h
BAR7	4F8h	4FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 9.7.3.2 Dual Cast Low BAR[0-7] Translation, Dual Cast High BAR[0-7] Translation

The 32-bit **Dual Cast Low and High BARx Translation** registers in Port 0 (refer to [Table 9-14](#)) specify the destination address of the Dual Cast Copy TLP for their corresponding Dual Cast BARs. When a Dual Cast Copy TLP is formed, the original TLP's Header Address bits are replaced with corresponding Address bits from these registers.

#### **Restrictions**

- Dual Cast BAR Translation addresses must be mapped to the Dual Cast Destination Port, as specified in the **Dual Cast Source Destination Port** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [510h](#)). (Refer to [Section 9.7.3.4](#).) While the original TLPs that are copied can be mapped to any PEX 8619 Port, copied TLPs from all Dual Cast BARs must be mapped to the Dual Cast Destination Port.
- Dual Cast BAR Translation address plus the Dual Cast BAR window size must never exceed the Address range mapped to the Dual Cast Destination Port.

**Table 9-14. Dual Cast High/Low BARx Translation Register Locations**

<b>Base Address Register</b>	<b>Port 0 Register Offset</b>	
	<b>Dual Cast Low Translation[31:0] (Lower 32 Bits)</b>	<b>Dual Cast High Translation[31:0] (Upper 32 Bits)</b>
BAR0	458h	45Ch
BAR1	470h	474h
BAR2	488h	48Ch
BAR3	4A0h	4A4h
BAR4	4B8h	4BCh
BAR5	4D0h	4D4h
BAR6	4E8h	4ECh
BAR7	500h	504h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.7.3.3 Dual Cast Low BAR[0-7] Setup, Dual Cast High BAR[0-7] Setup

The 32-bit **Dual Cast Low** and **High BARx Setup** registers in Port 0 (refer to [Table 9-15](#)) form a 64-bit value used to define the Dual Cast BAR window size, which increases in powers of 2 in size, starting from 1 MB.

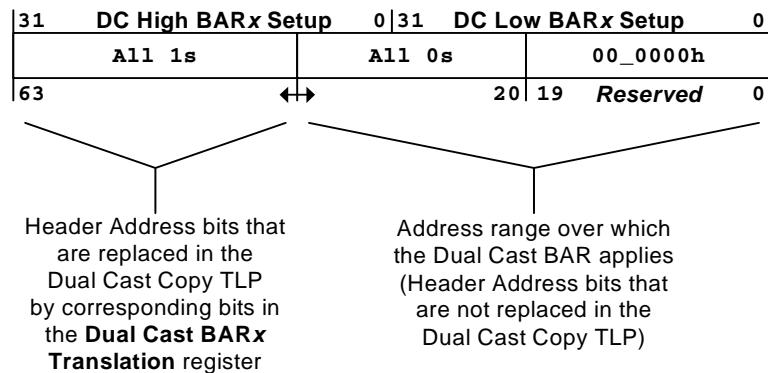
The Dual Cast BAR window size increases in powers of 2, starting from 1 MB. The address window size is determined by the quantity of Set bits, starting from bit 63 down to bit 20. For each bit that is a value of 1, the Copy TLP's Header Address bit is replaced by the corresponding Address bit in the **Dual Cast Low/High BARx Translation** registers, as illustrated in [Figure 9-10](#).

If bits [63:20] are all Set, the source window size is 1 MB. If bits [63:21] are all Set, and bit 20 is Cleared, the window size is 2 MB. When **Dual Cast High BARx Setup[31]** is Cleared, the corresponding Dual Cast BAR is disabled. [Table 9-16](#) lists example **Dual Cast Low/High BARx Setup** register values and the corresponding Dual Cast BAR Address window sizes.

**Table 9-15. Dual Cast High/Low BARx Setup Register Locations**

Base Address Register	Port 0 Register Offset	
	Dual Cast Low Setup[31:0] (Lower 32 Bits)	Dual Cast High Setup[31:0] (Upper 32 Bits)
BAR0	460h	464h
BAR1	478h	47Ch
BAR2	490h	494h
BAR3	4A8h	4ACh
BAR4	4C0h	4C4h
BAR5	4D8h	4DCh
BAR6	4F0h	4F4h
BAR7	508h	50Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 9-10. Dual Cast High/Low BARx Setup Register Example****Table 9-16. Dual Cast BARx Setup Register Address Window Sizing Examples**

Dual Cast High BARx Setup[31:0]	Dual Cast Low BARx Setup[31:20]	Dual Cast BARx Window Size
0000_0000h	XXXh	Disabled (default)
FFFF_FFFFh	FFFh	1 MB
FFFF_FFFFh	FFEh	2 MB
FFFF_FFFFh	000h	4 GB
FFFF_FFFEh	000h	8 GB

### Restrictions

- Dual Cast BAR Address windows must not overlap one another.
- Dual Cast Source Address windows 0 through 7 can be mapped to any Port (including the upstream Port and NT Port); however, only Write TLPs that enter the PEX 8619 through the designated Dual Cast Source Port are subject to Dual Casting. (Refer to [Section 9.7.3.4](#).)
- Both the original forwarded TLP **and** Dual Cast Copy TLP must be acknowledged on their respective egress Ports, before the original incoming TLP can be retired.
- Dual Cast BAR regions can be coded only as 64-bit BARs. For regions under 4 GB, Dual Cast High **BARx** = 0000\_0000h.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 9.7.3.4 Dual Cast Source Destination Port

The **Dual Cast Source Destination Port** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port; **Reserved** (RsvdP) for Dual Cast BAR Limit Lower register, offset **510h**) is used to specify the Dual Cast Source and Destination Ports. [Table 9-17](#) lists the bit definitions for this register. Bits not listed are **reserved** (zero).

**Table 9-17. Dual Cast Source Destination Port Register Definition  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port; Reserved (RsvdP) for Dual Cast BAR Limit Lower register, offset 510h)**

Bit(s)	Function
3:0	<b>Dual Cast Source Port #</b> Valid only when bit 8 ( <i>Dual Cast Source Port Enable</i> ) is Set. Specifies the source (ingress) Port Number upon which Dual Cast BARs are applied. Refer to register offset <b>510h[3:0]</b> for the associated Port Number values.
7:4	<b>Dual Cast Destination Port #</b> Specifies the destination (egress) Port Number to which Dual Cast BAR Translation addresses are mapped, and to which Dual Cast Copy TLPs will be queued. Refer to register offset <b>510h[7:4]</b> for the associated Port Number values.
8	<b>Dual Cast Source Port Enable</b> 0 = Dual Cast applies to Write TLPs entering any Port on the Dual Cast Source Port. In that case, the field [3:0] ( <i>Dual Cast Source Port #</i> ) value is “Don’t Care.” 1 = Dual Cast applies only to Write TLPs entering the PEX 8619, by way of the Dual Cast Source Port Number specified in field [3:0].

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 9.7.4 Dual Cast Programming Example

In this example, we consider a PEX 8619 configured as x8, x4, x4, with Port 0 being the upstream Port (connected to Root Complex), and Ports 1 and 3 configured as Transparent downstream Ports. Port 1 maps to a block of physical memory at Base address AAA0\_0000h (32-bit addressing). Port 3 maps to a block of memory at Base address BBB0\_0000h.

For this example, the switch must be programmed such that Write TLPs entering Port 0 and addressed to Port 1 (in the range AAA0\_0000h to AAAF\_FFFFh) are Dual Cast (copied) to the memory that is mapped to Port 3, starting at address BBB0\_0000h.

### 9.7.4.1 Register Programming Steps

1. Program the **Dual Cast BAR0** registers with the Base address of the memory on Port 1.  
**Dual Cast Low BAR0[31:0]**=AAA0\_000Ch (*Note: [19:0] are hardwired to 0\_0000h.*)  
**Dual Cast High BAR0[31:0]**=0000\_0000h
2. Program the **Dual Cast BAR0 Setup** registers to specify the Dual Cast BAR window size (1 MB).  
**Dual Cast Low BAR0 Setup[31:0]**=FFF0\_0000h (*Note: [19:0] are hardwired to 0\_0000h.*)  
**Dual Cast High BAR0 Setup[31:0]**=FFFF\_FFFFh
3. Program the **Dual Cast BAR0 Translation** registers with the address of the memory on Port 3.  
**Dual Cast Low BAR0 Translation[31:0]**=BBB0\_0000h (*Note: [19:0] are hardwired to 0\_0000h.*)  
**Dual Cast High BAR0 Translation[31:0]**=0000\_0000h
4. Program the **Dual Cast Source Destination Port** register bits, as follows:  
**Dual Cast Source Destination Port[3:0]**=0h (Dual Cast Source Port 0)  
**Dual Cast Source Destination Port[7:4]**=3h (Dual Cast Destination Port 3)  
**Dual Cast Source Destination Port[8]**=1 (Dual Cast on Port 0 only)

At this point, all Write TLPs entering the PEX 8619 by way of Port 0, having addresses in the range AAA0\_0000h to AAAF\_FFFFh, will be copied to memory at Base address BBB0\_0000h. To verify that Dual Cast is enabled, try Clearing memory at address BBB0\_0000h, then write a non-zero pattern to memory at address AAA0\_0000h. A read of memory at BBB0\_0000h should now show what was written at address AAA0\_0000h.

## 9.7.5 Dual Cast to a Non-Transparent Destination Port

When the Dual Cast Destination Port is configured as Non-Transparent (NT), the Dual Cast Copy TLPs do not use the same Address Translation mechanism as original unicast TLPs that are routed to that Port. Where a unicast TLP uses the Address Translation mechanism in the NT Port (NT BAR), Dual Cast Copy TLPs are sent directly to the NT Port's Egress queue. Therefore, in applications where the Dual Cast Destination Port is the NT Port, the **Dual Cast Low/High BARx Translation** registers must be loaded with the physical Base address of the destination memory in the NT address domain.

## 9.7.6 Error Reporting of Failed Dual Cast Cycles

If either the PCI address-routed destination device or the device connected to the Dual Cast Destination Port fails to return an ACK for the Dual Casted Memory Write, the standard PCI Express Correctable Errors (Replay Timer Timeout on first transmission attempt, Replay Rollover Status after four failed attempts) are reported.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova



## Chapter 10 Integrated DMA Engine – DMA Mode Only

### 10.1 Features

The integrated Direct Memory Access (DMA) engine includes the following features:

- Four independent channels – DMA Channels 0, 1, 2, and 3
- Up to 3.0 GB/s<sup>a</sup> throughput, per channel
  - Designed to 100% use a PCI Express x8 5.0 GT/s (Gen 2) Link
    - One channel saturates the Link, in one direction
    - Two channels saturate the Link, in both directions
  - Fully pipelined Data Read and Data Write
- Transfer between any Port, including the same Port
- 16-byte Descriptor
  - Standard 16-byte Descriptor holds 48-bit Source address, 48-bit Destination address, 27-bit Transfer Size, and five Control bits
  - Extension 16-byte Descriptor holds upper 16-bit Source address and upper 16-bit Destination address, to allow 64-bit addressing
- 64-bit addressing
- Off-Chip Descriptor mode
  - Hides Descriptor fetch latency from Data transfer
  - If all four channels are enabled, each channel can fetch from 1 to 64, 16-byte Descriptors
- On-Chip Descriptor mode
  - 1 to 256 Descriptors can be programmed directly, on-chip
  - Read of system RAM not necessary to fetch Descriptors
- Controls, per Descriptor
  - *Valid* bit to indicate that DMA can work on a Descriptor
  - Programmable interrupt
  - Constant Source and/or Destination address
- Controls, per channel
  - Programmable Read Size, from 4 to 2,048 bytes
  - Programmable spacing between Reads
  - Programmable Traffic Class, Relaxed Ordering, No Snoop
  - Run/Pause DMA
  - DMA Abort
- DMA versus non-DMA traffic priority policy, using Traffic Class (TC)
  - TC/VC mapping allows DMA to proceed to a different queue than a non-DMA queue
  - Requires the support of two VC on the Link, to acquire different queues

<sup>a</sup>. 3.0 GB/s includes all framing and protocol overhead, and is the Payload throughput for 64-byte Payload TLPs. Actual wire bandwidth approaches 4 GB/s, in each direction for a x8 Gen 2 Link.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

- DMA system debug
  - Done status per channel, visible to software
    - Current Descriptor
    - Actual Byte Count transferred
  - Optional Successful or Unsuccessful status, Invalidate Write Back, per Descriptor
  - Optional status (current Descriptor, Pause, Abort, and so forth), per channel
- DMA interrupt (Conventional PCI INTx or MSI) generated
  - Immediate Stop Done interrupt
  - If a Descriptor is directed to the end of another Descriptor
  - If an error occurs
  - If a Descriptor is invalid
  - Upon Pause Done
  - Upon Abort Done
  - 16 MSIs needed (four per channel)
  - 8 DMA interrupt Vectors (two per channel)
- Predictable Ordering – Per channel, in Descriptor order, for
  - DMA Reads
  - DMA Writes
  - DMA Completions
  - DMA interrupts
- DMA partial Completion combining
  - Combine partial Completions for one MRd tag
  - 64 tags
    - **One-Channel mode** – 63 tags for MRd, and one tag dedicated to Descriptor Read
    - **Four-Channel mode** – 60 tags for MRd (15 per channel), and four tags reserved for Descriptor Read (one per channel)
  - MRd Size programmable, up to Maximum Payload Size
  - MWr Size = MRd Size

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 10.2 Terminology

Table 10-1 lists miscellaneous DMA and PCI Express-specific terminology used in this chapter.

**Table 10-1. Miscellaneous DMA- and PCI Express-Specific Terminology**

Terms and Abbreviations	Description
Data MRd	Data Memory Read. Reading from the Source address, to gather data to transfer.
Data MWr	Data Memory Write. Writing to the Destination address, to transfer Read data.
Descriptor	16 bytes of information that describes the DMA operation or instruction, <i>such as</i> the Source address to read, Destination address to write, and total Transfer Size.
DMA	Direct Memory Access.
DMA Ring	Physical location where the DMA Descriptors reside. When the last Descriptor is done, the pointer automatically moves to the location of the first Descriptor, creating a DMA Descriptor Ring.
Partial Completion	In PCI Express, a single Read can have one or more Completions with data. If more than one Completion is returned, they are referred to as <i>Partial Completions</i> .

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 10.3 General Description

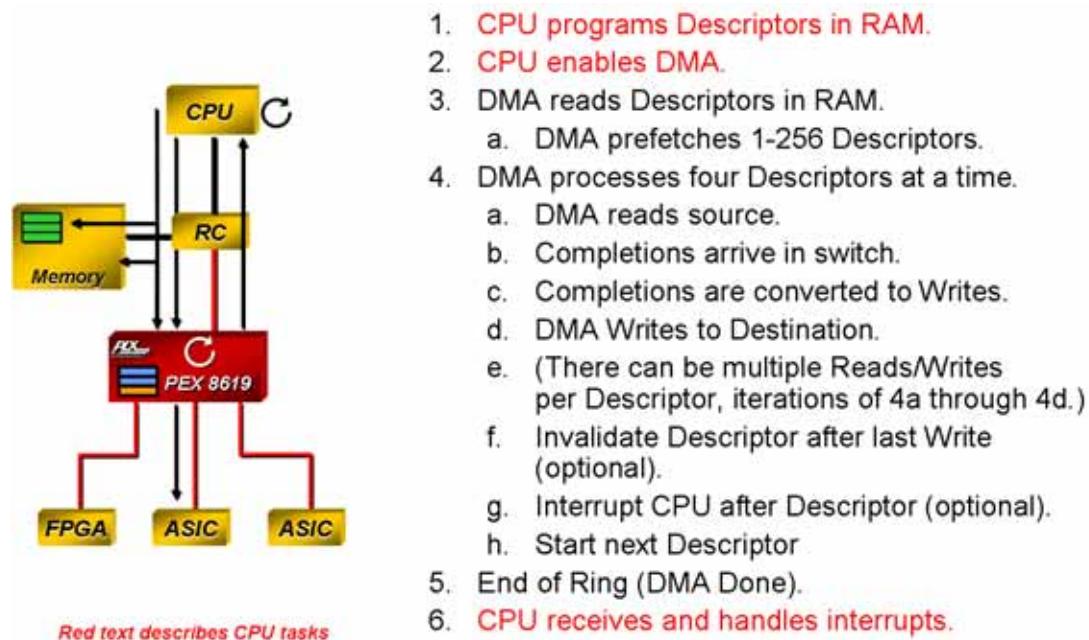
DMA is a method used for moving data, without burdening the CPU. The CPU sets up DMA Descriptors to move data, turns On the DMA engine, and is then free to perform other processes while the DMA engine moves the data. The DMA engine can also interrupt the CPU and provide progress status at programmable intervals.

The PEX 8619's four DMA channels allow different, co-existing DMA transfers to run concurrently. A Descriptor provides the DMA channel with the necessary details for performing a transfer, after a DMA channel is turned On. The Descriptor is either programmed directly on the PEX 8619 ahead of time (DMA On-Chip Descriptor mode; refer to [Section 10.6.3](#)), or loaded from Host RAM (DMA Off-Chip Descriptor mode; refer to [Section 10.6.4](#)). If the Descriptor is loaded from Host RAM, multiple Descriptors can be loaded simultaneously, hiding the Descriptor Fetch time from the DMA movement. After the Descriptor is loaded, the Data Movement phase begins.

Each Descriptor includes a Source address, Destination address, Transfer Size, and Interrupt policy. Data is read from the Source address, by sending one or more Memory Reads at a programmable rate. The Read Completions are intercepted by the DMA engine at the Read Completion receiving Port (point of ingress) and turned into Writes to the Destination address. This process continues, until the Transfer Size is met, indicating that the Descriptor is done. The DMA engine can process and work on four Descriptors at a time. If one channel is enabled, the DMA engine will work on four Descriptors from that channel. If four channels are enabled, however, the DMA engine will concurrently work on only one Descriptor from each channel.

[Figure 10-1](#) illustrates the DMA operation sequence for a single channel. In this example, the DMA engine moves data from the main RAM to an endpoint. The CPU is used only in steps 1, 2, and 6; the remaining steps are performed by the DMA engine.

**Figure 10-1. DMA Operation Sequence**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 10.4 DMA Usage Models

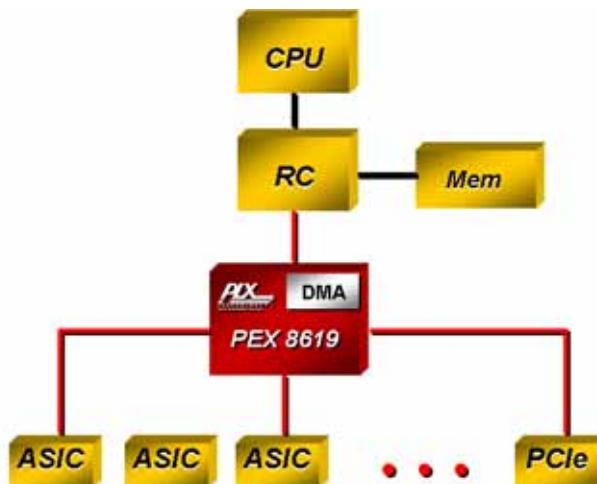
To help balance its workload, the CPU sets up Descriptors, then writes a 1 to the **DMA Channel x Control/Status** register *DMA Channel x DMA Start* bit(s) (DMA of Upstream Port, offsets **238h[3]**, **338h[3]**, **438h[3]**, and/or **538h[3]**), to initiate the Data transfer. The Source address, Destination address, and Transfer Size describe the data movement. The DMA Controller generates and sends interrupts regarding DMA transfer's progress to the Host.

The PEX 8619's DMA supports transfers anywhere within its Address map (32- or 64-bit Memory space). The DMA engine can write to, and read from, any accessible upstream Memory space.

### 10.4.1 Statistics Gathering

Figure 10-2 illustrates a Host CPU that needs to process statistics gathered on  $M$  downstream ASICs. Rather than have the CPU read all the ASICs, the Host CPU sets up a DMA engine to periodically read the ASICs. By setting up  $N$  sets of  $M$  Descriptors, the CPU need only look at the statistics in Main memory, after  $N \times M$  Descriptors are done.

**Figure 10-2. Statistics Gathering Usage Model**



## 10.4.2 Peer-to-Peer Transfer

The DMA engine can move data from one downstream peer to another, without involving the chipset, thus avoiding upstream bottlenecks.

## 10.4.3 Move Data in Main Memory

The DMA engine can read Main memory, then write back to a different location, using either of the following methods:

- Gather scattered data in Main memory, then combine the data into one large block
- Gather a large chunk of data in Main memory, then break the data into smaller pieces

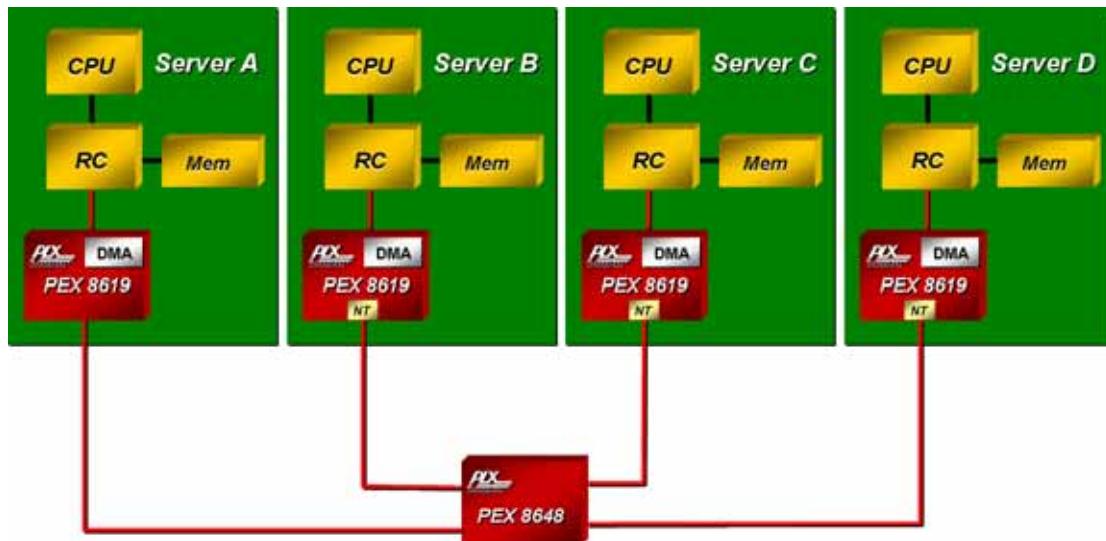
## 10.4.4 Endpoint

The DMA engine combined with the PEX 8619's Non-Transparent (NT) Port provides a powerful building block for an endpoint. Any device with a native PCI Express interface can be attached to the endpoint, then distributed across multiple Ports, to scale to larger capacity.

## 10.4.5 Interprocessor Communication

The DMA engine with the PEX 8619's NT Port can also communicate across two Host domains. Provided that a global Address table is available, each Host can write to or read from another Host's memory. (Refer to [Figure 10-3](#).)

**Figure 10-3. DMA and NT for Interprocessor Communication Usage Model**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 10.5 Software Interface

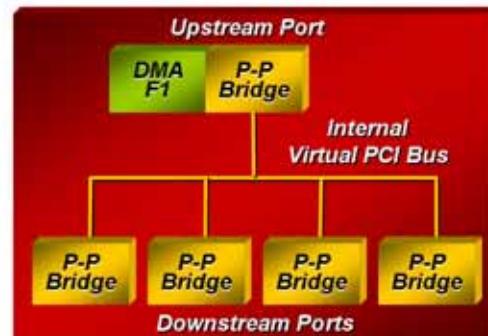
Figure 10-4 illustrates a software view of the PEX 8619 with DMA. The PEX 8619 is composed of multiple PCI-to-PCI (P-P) bridges. The DMA Controller is seen as a Type 0 endpoint on Function 1 (F1) of the upstream device.

By having DMA as a separate function, software loads a driver for the DMA engine during discovery. Software can reach the DMA Control registers and on-chip registers, using Memory-Mapped and/or Configuration Space Register (CSR) accesses (limited to 4 bytes for Reads and Writes). A complete list of DMA-specific registers is provided in Chapter 18, “DMA Interface Registers – DMA Mode Only.”

**Figure 10-4. Software View of PEX 8619 with DMA**

### PEX 8619 is a multi-function device

- Function 0 → P-to-P bridge
  - Transparent Switch model
  - No driver required
- Function 1 → DMA endpoint
  - Type 0 Configuration Header
  - Requires DMA driver  
(provided by PLX) \*
- DMA registers are memory-mapped for easy access



\*At the time of this writing, only Microsoft Windows XP, Microsoft Vista, and Linux are supported.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 10.6 DMA Descriptors

Descriptors direct the work processed by the DMA engine. The DMA engine implements a Descriptor Ring format. This implementation allows the Descriptor pointer to automatically point to the first Descriptor's location, after the last Descriptor is processed. [Figure 10-5](#) illustrates how Descriptors are stored within a Ring.

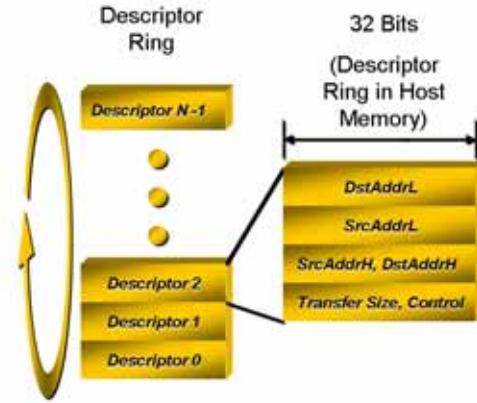
The Descriptors can be loaded directly onto the PEX 8619 ([DMA On-Chip Descriptor Mode](#)), or fetched during DMA runtime ([DMA Off-Chip Descriptor Mode](#)). Regardless of the mode, the Descriptors have the same format.

Each Descriptor is 16 bytes in length, and has a *Valid* bit, which is used to indicate whether the Descriptor contains valid data. If the Descriptor is invalid, the Descriptor remains unused.

The two Descriptor formats – Standard and Extended – are described in the sections that follow.

**Figure 10-5. Descriptor Ring**

- Descriptors are instructions for DMA
  - Written by CPU
  - Stored in a ring in Host Memory  
-OR-
  - Stored internal to the PEX 8619
- 16-byte standard format
  - Supports 32-bit Addressing
  - Control/Status information
- 16-byte extended format
  - Supports 64-bit Addressing
  - Control/Status information



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 10.6.1 Standard Descriptor Format

The Standard Descriptor format is the only Descriptor that requires programming to move data.

[Figure 10-6](#) illustrates and describes the Standard Descriptor format.

**Figure 10-6. Standard Descriptor Format**

	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>DWord 3</b>	Lower SrcAddr [31:0]																																			
<b>DWord 2</b>	Lower DstAddr [31:0]																																			
<b>DWord 1</b>	SrcAddr[47:32]																									DstAddr[47:32]										
<b>DWord 0</b>	DV	D	RC	WC	DF	Transfer Size [26:0]																														

### DWord 0 Descriptor Control

Bit 31 = *Descriptor Valid*.

Bit 30 = *Interrupt when Done with Descriptor*.

Bit 29 = *Hold Read Address Constant*.

Bit 28 = *Hold Write Address Constant*.

Bit 27 = *Descriptor Format* (0 = Standard).

*SrcAddr[1:0]* must match *DstAddr[1:0]* – both fields can be any value, as long as both values are the same.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 10.6.2 Extended Descriptor Format

The Extended Descriptor format provides a method for changing the upper 16 bits of the Source and/or Destination address, using Descriptors, which enables the entire 64 bits of PCI Express Address space. Once programmed, the upper 16 bits remain the same for all subsequent Descriptors (until a new extended Descriptor is received). The Extended Descriptor also allows for the Descriptor Ring to move.

[Figure 10-7](#) illustrates and describes the Extended Descriptor format.

**Figure 10-7. Extended Descriptor Format**

	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>DWord 3</b>	<i>Reserved</i>																										SrcPortNum Mask					
<b>DWord 2</b>	<i>Reserved</i>																										DstPortNum Mask					
<b>DWord 1</b>	SrcAddr[63:48]																										DstAddr[63:48]					
<b>DWord 0</b>	DV	SV	DV	<b>Rsvd</b>	DF	<i>Reserved</i>																										

### DWord 0 Descriptor Control

Bit 31 = *Descriptor Valid*.

Bit 30 = *Upper SrcAddr Valid*.

Bit 29 = *Upper DstAddr Valid*.

Bit 28 = **Reserved**.

Bit 27 = *Descriptor Format* (1 = Extended).

Bits [26:24] = **Reserved**.

The Extended Descriptor must be valid (in the same way that the Standard Descriptor must be valid). When the Extended Descriptor is valid, it provides two functions, each of which can be enabled separately, by way of the corresponding *Valid* bit:

- **Bit 30 = 1** – The Upper Source Address bits, [63:48], takes this new value. All subsequent Descriptors will use the new Upper Source Address bits.

**Bit 30 = 0** – There is no change from the previous *SrcAddr[63:48]* bits value stored.  
If there was never a valid *SrcAddr Valid* bit, *SrcAddr[63:48]* will be Cleared.

- **Bit 29 = 1** – The Upper Destination Address bits, [63:48], can be modified. All subsequent Descriptors will use the new Upper Destination Address bits.

**Bit 29 = 0** – There is no change from the previous *DstAddr[63:48]* bits value stored.  
The initial value for *DstAddr[63:48]* is 0.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

The *SrcPortNum Mask* bits [15:0] provide a means (one-hot mask) to check the routing for the generated DMA Memory Read Request. If one of the *SrcPortNum Mask* bits is Set, Data Reads are not allowed to be sent out from that Port. The one-hot mask applies to DMA Data Reads only – it does not apply to Descriptor Read/Write. *For example*, if bit 0 is Set, Read Requests are not allowed to exit on Port 0. Instead, any Read Requests targeted to exit on Port 0 will be dropped, and the DMA transaction will be terminated with a Source Port error.

The *DstPortNum Mask* bits [15:0] provide a means (one-hot mask) to check the routing for the DMA Memory Write Request generated. The one-hot mask applies to DMA Data Writes only – it does not apply to Descriptor Read/Write. If one of the *DstPortNum Mask* bits is Set, DMA Data Writes are not allowed to be sent out from that Port, in the same way as with the *SrcPortNum Mask* bits.

### 10.6.3

### DMA On-Chip Descriptor Mode

In DMA On-Chip Descriptor mode ([DMA Global Control](#) register *DMA On-Chip Descriptor Mode* bit (DMA of Upstream Port, offset [1FCh\[2\]](#)) is Set), an on-chip Descriptor RAM is visible to software. Any address in the Descriptor RAM can be read from or written to by software, four bytes at a time. The 256, 16-byte entries comprise 4 KB of CSR space. The 4-KB space starts at address offset 1000h. The DMA Descriptor RAM can be used as a Scratchpad RAM, regardless of whether DMA is enabled.

In On-Chip Descriptor mode, all Descriptors are directly programmed into this RAM, removing the need for the DMA engine to access Local memory to acquire Descriptors. (By comparison, in Off-Chip Descriptor mode, Descriptors from off-chip are copied onto this RAM, for internal consumption.)

While running On-Chip Descriptor mode, the *Valid* bit for the Descriptors is not Cleared. The Descriptor Ring Size indicates the quantity of on-chip Descriptors. When the last Descriptor is read, the DMA engine either stops, pauses, or delays, before looping back to the first Descriptor.

## 10.6.4 DMA Off-Chip Descriptor Mode

In DMA Off-Chip Descriptor mode (**DMA Global Control** register *DMA On-Chip Descriptor Mode* bit (DMA of Upstream Port, offset **1FCh[2]**) is Cleared), the Descriptors are read from off-chip. The DMA channel must be programmed with a Descriptor Ring Base address and Descriptor Ring Size, before being activated.

In the Ring, the Descriptors are read, in order, starting at the Base address and moving up the address from there. When the end of the Ring is reached, the Descriptors continue being fetched from the start of the Ring again, in an endless loop. The delay of the Descriptor prefetch of the Descriptor Ring wraparound is programmable, by way of the **DMA Channel x Control/Status** register *DMA Channel x DMA Ring Delay Period* field(s) (DMA of Upstream Port, offsets **238h[15:13]**, **338h[15:13]**, **438h[15:13]**, and/or **538h[15:13]**).

This mode can be used to prefetch Descriptors (1 to 256) that are not yet active. By prefetching the Descriptors, the DMA Data transfer can remain active 100% of the time. Without prefetching, after finishing one Descriptor, the DMA would have to wait for 300+ ns to fetch the next Descriptor, essentially losing bandwidth while waiting.

DMA Descriptors are prefetched four at a time<sup>a</sup> (4 x 16 bytes = 64 bytes common Cache Line Size). However, if the Prefetch limit is programmed to 1, only one Descriptor is prefetched at a time (1 x 16 bytes = 16 bytes common Cache Line Size).

If any Descriptor is invalid, then that Descriptor and any subsequent Descriptors are discarded. If enabled, an Invalid Descriptor interrupt is generated. Furthermore, the DMA Descriptor Invalid Status flag is Set (reported in the **DMA Channel x Control/Status** register *DMA Channel x DMA Descriptor Invalid Status* bit(s) (DMA of Upstream Port, offsets **238h[8]**, **338h[8]**, **438h[8]**, and/or **538h[8]**)), which prevents prefetching of Descriptors until the flag is Cleared.

*For example*, suppose that Descriptors at 400h, 410h, 420h, and 430h are prefetched. The Descriptor at 400h is valid, but the Descriptor at 410h is found to be invalid. As a result, Descriptors at 410h, 420h, and 430h are all discarded. The next Descriptor to prefetch is programmed to 410h, and the DMA Descriptor Invalid Status flag is Set.

---

a. For example, if the Descriptor address is not 64-byte-aligned, it is aligned first, prior to being prefetched. Descriptors must be at 16-byte-aligned Address Boundary spaces. If the Descriptor is read at address 0A50\_0010h, then only three Descriptors are fetched, to access the 64-byte-aligned address of 0A50\_0040h.

## 10.6.5 DMA Descriptor Writeback

When the DMA engine transmits the last byte to write, the Descriptor is done. The DMA engine can then be optionally programmed to write back to the Descriptor, to Clear the *Valid* bit (bit 31) and invalidate the Descriptor. The writeback is a 4-byte Write to the bottom of the Descriptor.

The DMA engine stops if it moves around the Descriptor Ring and encounters an invalid Descriptor.

## 10.7 DMA Alignment

Any Start address can be given to the Source address. However, the Destination address' two least significant bits (lsbs) must match the Source address' lsbs. This requirement – matching the two lsbs in the Source and Destination addresses – significantly simplifies the hardware.

For performance, two other alignment cases should be considered – 4-KB and Maximum Payload Size.

PCI Express does not allow Requests to cross a 4-KB Address Boundary space. For DMA, this affects the quantity of PCI Express accesses required for Reads and Writes. In the worst case, there can be three extra Reads and Writes per 4 KB, if the initial alignment is not optimal.

The PEX 8619's DMA engine always attempts to read the Maximum Programmed Read Size, which can be up to the system's Maximum Payload Size (MPS). If a Read does not start at an aligned MPS Address Boundary space, per Descriptor, there can be a Start and End Read that is less than the MPS, thereby impacting performance to a small degree.

*For example*, if a 256-byte (100h) Read starts at address A040\_0FF0h, there must be two different Read Requests – a 16-byte (10h) Read to reach the 4-KB Address Boundary space A040\_1000h, and then a 240-byte (F0h) Read to complete the transfer. Compare this to a 256-byte Read at the MPS-aligned address of A040\_0E00h – one Read is all that is needed for the entire 256 bytes. Additionally, the DMA engine can issue Read Requests with a Read Request smaller than the MPS, to align the Completions to a 64-byte boundary. This is necessary to combine the partial Completions received, into Writes with a Transfer Size equal to the MPS.

Writes have the same limitations for crossing the 4-KB Address Boundary space. To simplify the hardware, one Write is created for each Read. Therefore, Read Sizes must consider both the Read and Write 4-KB Address Boundary spaces.

To maximize performance, the Read and Write addresses should be 4-KB aligned (same 12 lsbs). Furthermore, to maintain maximum throughput, Reads should be Maximum Read Size (up to the MPS)-aligned.

Table 10-2 describes a system with a 256-byte MPS, as well as DMA channel behavior for a given Transfer Size with an unaligned Start address on the source and destination. **DMA Channel x Current Descriptor Transfer Size** register *DMA Channel x Current Descriptor Transfer Size* field (DMA of Upstream Port, offsets 228h[26:0], 328h[26:0], 428h[26:0], and/or 528h[26:0]) value(s) indicate the quantity of bytes transferred. As can be observed from the table, there are cases in which the amount transferred is less than the MPS. This relates to the alignment to 64-byte Address Boundary space, as well as not crossing the 4-KB Address Boundary space. The table's *Action* column provides additional information regarding instances in which the amount transferred is less than 256 bytes. All Reads and Writes (in the table) are the same size.

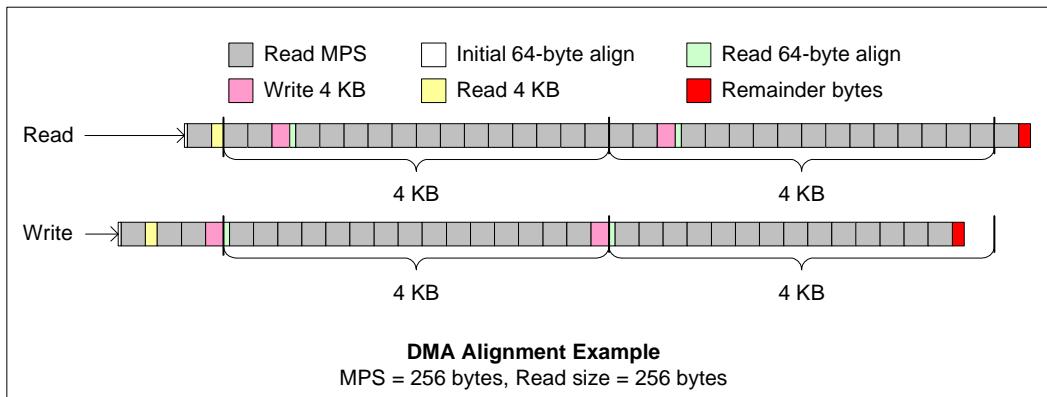
Figure 10-8 illustrates the data presented in Table 10-2. The upper portion illustrates the Read stream, as well as the 4-KB Address Boundary spaces. Similarly, the lower portion illustrates the Write stream, as well as the 4-KB Address Boundary spaces. In this example, the Read 4-KB Address Boundary space precedes the Write 4-KB Address Boundary space.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 10-2. DMA Transfer with Unaligned Addresses**

- System MPS = 256 bytes
  - Maximum Read Size = 256 bytes
- Source Address = 4E44h
  - Bits [1:0] = 00b
- Destination Address = 0BA8h
  - Bits [1:0] = 00b
- Transfer Size = 8,938 Bytes

Action	Read Address	Write Address	Bytes Transferred	Bytes Remaining
Align 64B	4E44h	0BA8h	60	8,938
	4E80h	0BE4h	256	8,878
Read 4 KB	4F80h	0CE4h	128	8,622
	5000h	0D64h	256	8,494
	5100h	0E64h	256	8,238
Write 4 KB	5200h	0F64h	156	7,982
Align 64B	529Ch	1000h	36	7,826
	52C0h	1024h	256	7,790
Read 4 KB	5FC0h	1D24h	64	4,462
	6000h	1D64h	256	4,398
	6100h	1E64h	256	4,142
Write 4 KB	6200h	1F64h	156	3,886
Align 64B	629Ch	2000h	36	3,730
	62C0h	2024h	256	3,694
	6EC0h	2C24h	256	622
Read 4 KB	6FC0h	2D24h	64	366
	7000h	2D64h	256	302
Remainder	7100h	2E64h	46	46

**Figure 10-8. DMA Alignment Example**

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 10.8 Performance Considerations

The performance goal for the PEX 8619's DMA engine is 100% Link usage, in both directions for a x8 Gen 2 Link. With an average payload of 64 bytes, the PEX 8619 can sustain up to 3.0 GB/s per direction, per DMA channel.

This section consolidates the various methods for improving performance, into the following list.

*For optimal performance:*

- Set the MPS as high as possible (up to 2,048 bytes for the PEX 8619)
- Set the Maximum Read Size to the same value as the MPS
- Do one of the following:
  - Place Descriptors in on-chip Descriptor RAM, –or–
  - Program a large Transfer Size, per Descriptor, to minimize Descriptor fetching
- Align Source address and Destination address to MPS Address Boundary space
- Align Source address and Destination address to 4-KB Address Boundary space

For cases in which such high throughput hinders overall system performance, the DMA engine also implements the following alternative methods of restricting performance:

- [DMA Maximum Bandwidth](#)
- [DMA Maximum Round-Trip Read Time](#)
- [DMA and Non-DMA Policies](#)

### 10.8.1 DMA Maximum Bandwidth

A DMA channel's maximum bandwidth can be controlled, by Setting the Maximum Read Size and programming spacing between subsequent Reads. The spacing is Set, per channel, and is programmed in units of 4 ns.

*For example*, assume that the Maximum Read Size for a DMA channel is programmed to 256 bytes. Also, assume that the application requires the DMA channel to limit its bandwidth to 100 MB/s. In this case, the DMA channel must limit Data transfers to 256 bytes, every 2.56 µs (256 bytes/100 MB/s). The 2.56 µs period can be further broken down, into 4-ns units. The resulting value of 640 (in 4-ns units; 2.56 µs/4 ns) is the value used to configure the **DMA Channel x Bandwidth** register **DMA Channel x DMA Clocks to Transfer** field(s) (DMA of Upstream Port, offsets [22Ch\[15:0\]](#), [32Ch\[15:0\]](#), [42Ch\[15:0\]](#), and/or [52Ch\[15:0\]](#)). As a result, the DMA channel does not consume more than 100 MB/s for its Data transfers. However, if the remainder of the DMA engine is not optimal (refer to [Section 10.8](#)), the DMA engine might fall short of the desired bandwidth.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 10.8.2 DMA Maximum Round-Trip Read Time

It takes a round-trip delay, to receive Completion data after issuing a Read. As this time elapses, more data must be requested, to maintain maximum continuous throughput. Additional data can be requested by issuing larger, or more, Read Requests.

The Read Request Size sent by the DMA engine is programmable, and can be up to the system MPS. Per the *PCI Express Base r2.0*, all PCI Express systems support a minimum MPS of 128 bytes. Some systems can support up to the 2,048-byte MPS supported by the PEX 8619.

The quantity of DMA Reads depends upon the quantity of available DMA Read tags. Each DMA Read Request consumes a unique tag. The tag is used to pair the Read Completion with the original Read Request, to ensure correct sequencing of the Writes.

A total of 64 DMA Read tags are available. Each channel uses one tag for a Descriptor fetch, leaving the remaining tags available for Data Reads. The following summarizes the total quantity of available tags, based upon the quantity of active channels:

- **1 channel** – 63 Data Read tags + 1 Descriptor tag
- **2 channels** – (31 Data Read tags + 1 Descriptor tag) x 2
- **3 or 4 channels** – (15 Data Read tags + 1 Descriptor tag) x 4

The maximum round-trip time that a DMA channel can cover, while maintaining a fully used Link, is provided by the following formula:

$$\text{MaxRoundTrip} = (20 \text{ bytes} + \text{Read Request Size}) \times \text{MaxTags} / \text{LinkByteRate}$$

With a Maximum Payload Size of 256 bytes, two DMA channels, and two x8 PCI Express Links, the largest round-trip time that the DMA can cover, while maintaining 100% PCI Express usage, is  $(276) \times 31 / 4 \text{ GB/s} = 2,139 \text{ ns}$ .

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 10.8.3 DMA and Non-DMA Policies

The PEX 8619 can potentially have DMA and non-DMA traffic running concurrently. For each traffic type, there are several available methods, for controlling the Link usage ratio.

The first method – controlling the maximum bandwidth – is discussed in [Section 10.8.1](#). However, if it is necessary for the DMA engine to consume all available bandwidth, but on a lower priority than non-DMA traffic, use one of the following options:

- DMA Traffic Class
- DMA Relaxed Ordering
- DMA No Snoop

Within one channel, different DMA Requests (Descriptor Read/Write and Data Read/Write) can have programmable Relaxed Ordering and No Snoop.

#### 10.8.3.1 DMA Traffic Class

Each DMA channel can be assigned a unique Traffic Class (TC). PCI Express allows eight different Traffic Classes (TC[7-0]), none of which are ordered against one another. Typical default non-DMA traffic uses only TC0. Setting a DMA channel to a non-zero Traffic Class provides the opportunity for different Quality of Service (QoS).

The PEX 8619 supports two Virtual Channels (VCs) – VC0 and VC1 – per physical Link. A TC can be mapped to any VC (with the exception of TC0, which can map only to VC0). The PEX 8619 allows three schemes for scheduling VC0 and VC1 traffic, by way of the **DMA Channel x Control/Status** register *DMA Channel x DMA Traffic Class* field(s) (DMA of Upstream Port, offsets [238h\[21:19\]](#), [338h\[21:19\]](#), [438h\[21:19\]](#), and/or [538h\[21:19\]](#)):

- VC0 higher priority
- VC0 lower priority
- VC0 equal priority

By putting DMA traffic onto VC1, VC1's traffic can be separated from VC0 (default) traffic, for any Link that supports two VCs.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 10.8.3.2 DMA Relaxed Ordering

Each DMA channel can enable Relaxed Ordering on its Request TLPs. When the **DMA Channel x Control/Status** register Relaxed Ordering-related bits (DMA of Upstream Port, offsets [238h\[25:22\]](#), [338h\[25:22\]](#), [438h\[25:22\]](#), and/or [538h\[25:22\]](#)) are Set, the corresponding DMA Descriptor and Data Read and Write Requests will have the RO bit in the TLP Header Set, to take advantage of the PCI Express Relaxed Ordering rules.

Every PCI Express TLP Header has a *Relaxed Ordering* bit. If this bit is Set, Ordering rules are relaxed, and the Relaxed Ordering TLP can pass (or be passed) by another TLP.

The PEX 8619 looks only at the *Relaxed Ordering* bit for Completions. Completions with the *Relaxed Ordering* bit Set are sent to a different Egress queue than other traffic, and can therefore be scheduled at an earlier time.

For systems with only one VC, and where it is necessary to have DMA consume all available bandwidth while running at a lower priority than non-DMA traffic, the non-DMA traffic can Set the *Relaxed Ordering* bit.

### 10.8.3.3 DMA No Snoop

Each DMA channel can also program and enable the PCI Express *No-Snoop (NS)* bit to be Set in its TLP Headers. When the **DMA Channel x Control/Status** register No Snoop-related bits (DMA of Upstream Port, offsets [238h\[29:26\]](#), [338h\[29:26\]](#), [438h\[29:26\]](#), and/or [538h\[29:26\]](#)) are Set, the associated Descriptor and Data Read or Write Request TLPs have the *NS* bit Set in their PCI Express Headers, allowing applications that need this mechanism to enable No Snoop.

## 10.9 DMA Software Access

This section describes how to use software to access the DMA engine.

### 10.9.1 Configure a DMA Transfer

To configure a DMA transfer, the Descriptors must first be programmed. Descriptors can be programmed on- or off-chip, in local RAM.

With the Descriptors in place, the PEX 8619 must then be configured for DMA. After the **DMA Global Control** register **DMA Channel Mode** field (DMA of Upstream Port, offset **1FC<sub>h</sub>[1:0]**) is used to select the quantity of DMA channels (one, two, or four; three channels is not an option), each channel can be independently programmed and started at any time.

With the channel ready to start, Set the **DMA Channel x Control/Status** register **DMA Channel x DMA Start** bit(s) (DMA of Upstream Port, offsets **238<sub>h</sub>[3]**, **338<sub>h</sub>[3]**, **438<sub>h</sub>[3]**, and/or **538<sub>h</sub>[3]**), to turn On the DMA engine.

Table 10-3 lists the registers specific to programming the DMA Channels.

**Table 10-3. DMA Channel Programmable Registers**

Register	DMA Channel and Offset			
	0	1	2	3
DMA Channel x Source Address Lower	200h	300h	400h	500h
DMA Channel x Source Address Upper	204h	304h	404h	504h
DMA Channel x Destination Address Lower	208h	308h	408h	508h
DMA Channel x Destination Address Upper	20Ch	30Ch	40Ch	50Ch
DMA Channel x Transfer Size	210h	310h	410h	510h
DMA Channel x Descriptor Ring Address Lower	214h	314h	414h	514h
DMA Channel x Descriptor Ring Address Upper	218h	318h	418h	518h
DMA Channel x Next Descriptor Ring Address Lower	21Ch	31Ch	41Ch	51Ch
DMA Channel x Descriptor Ring Size	220h	320h	420h	520h
DMA Channel x Last Descriptor Address Lower	224h	324h	424h	524h
DMA Channel x Last Descriptor Transfer Size	228h	328h	428h	528h
DMA Channel x Bandwidth	22Ch	32Ch	42Ch	52Ch
DMA Channel x Maximum Prefetch Limit	234h	334h	434h	534h
DMA Channel x Control/Status	238h	338h	438h	538h
DMA Channel x Interrupt Control/Status	23Ch	33Ch	43Ch	53Ch
DMA Channel x Error Header Log 0	240h	340h	440h	540h
DMA Channel x Error Header Log 1	244h	344h	444h	544h
DMA Channel x Error Header Log 2	248h	348h	448h	548h
DMA Channel x Error Header Log 3	24Ch	34Ch	44Ch	54Ch
DMA Channel x Device-Specific Error Status	250h	350h	450h	550h
DMA Channel x Maximum Outstanding Read/Write Request	254h	354h	454h	554h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 10.9.2 DMA Pause

The DMA engine can be paused at any time, by Setting the **DMA Channel x Control/Status** register *DMA Channel x DMA Graceful Pause* bit(s) (DMA of Upstream Port, offsets [238h\[0\]](#), [338h\[0\]](#), [438h\[0\]](#), and/or [538h\[0\]](#)). This allows all current active Descriptors to complete (be done). No new Descriptors are prefetched when the DMA engine is paused.

After the current Descriptors are done, a Pause Done status flag (reported in the **DMA Channel x Control/Status** register *DMA Channel x DMA Graceful Pause Done Status* bit(s) (DMA of Upstream Port, offsets [238h\[9\]](#), [338h\[9\]](#), [438h\[9\]](#), and/or [538h\[9\]](#), is (are) Set)) communicates to software that the DMA channel is paused. Software can then read the next Descriptor pointer, to determine how far the DMA transfer has progressed.

To resume DMA operation from the Pause condition, Clear the *DMA Channel x DMA Graceful Pause* bit(s).

## 10.9.3 DMA Abort

In some cases, the DMA transfer might need to be aborted. *For example*, if a large transfer no longer needs to be completed, software can abort the DMA transfer, by Setting the **DMA Channel x Control/Status** register *DMA Channel x DMA Abort* bit(s) (DMA of Upstream Port, offsets [238h\[1\]](#), [338h\[1\]](#), [438h\[1\]](#), and/or [538h\[1\]](#)). When the *DMA Abort* bit is Set, no additional Requests are issued. This means that there are no more Descriptor fetches, nor Data Reads or Writes. Because no Descriptors are done, neither Descriptor interrupts nor writebacks are generated.

One loose end is waiting for all outstanding Reads to be done. To avoid having surprise Completions later, the DMA engine flushes all outstanding Read Requests on the scoreboard. It is not necessary to wait for the Received Completion data, because all Received Completion data is discarded.

An Abort Done status flag (reported in the **DMA Channel x Control/Status** register *DMA Channel x DMA Abort Done Status* bit(s) (DMA of Upstream Port, offsets [238h\[10\]](#), [338h\[10\]](#), [438h\[10\]](#), and/or [538h\[10\]](#), is (are) Set)) indicates that the Abort is done. This means that all outstanding Reads are done and the DMA status is up-to-date. Software can read the last Descriptor written, as well as the remaining bytes to be transferred, to determine how far the DMA transfer has progressed.

Software then Clears the Abort Done status flag and *DMA Abort* bit, before attempting to start another DMA transfer on the channel.

## 10.9.4 DMA Runtime Status

Once the DMA engine is running, runtime status can be viewed, without having to pause or abort the DMA transfer. This is accomplished by looking at the following information, which is recorded by the on-chip registers:

- Last Descriptor prefetched
- Last Descriptor finished
- Current Transfer Count
- Last Descriptor Transfer Size

In some cases, the CPU cannot probe down to the PEX 8619, to determine status, because it takes longer for the CPU to access the PEX 8619's internal registers than it takes to read from the local RAM. To support these cases, DMA status can be written to Local RAM, upon the Completion of each Descriptor.

Additionally, each Descriptor has a *Last Descriptor Transfer Size* bit. When Set, a Completion of the selected Descriptors generates an interrupt to the Host. Interrupts are grouped into two types for each channel – normal and error. The Descriptor Done interrupt is an example of a normal types of interrupt.

## 10.9.5 DMA Routing Check

The DMA engine can perform a routing authorization check, for Data Reads and Data Writes. A Data Read can be allowed or disallowed to read upstream, downstream, or out the NT Port. Similarly, a Data Write can be allowed or disallowed to write upstream, downstream, or out the NT Port.

If a Descriptor attempts to read from or write to a disallowed route, the TLP is dropped, and the Descriptor is skipped.

For further details regarding the interrupt options and errors generated, refer to [Section 10.9.6](#).

## 10.9.6 DMA Error Header Logging

The DMA engine has the capability, per channel, to log the Header of the first packet that causes difficulty. Difficulty is considered to be anything that causes the transfer to fail, including:

- Link is down
- Disallowed direction

Standard PCI Express errors (*such as* a malformed packet) can be trapped outside the DMA engine. If a Read Completion results in a malformed packet, the DMA engine reports a Read timeout, while the standard PCI Express logic reports the malformed packet. The Read Header and Malformed Packet Header are both logged.

## 10.9.7 DMA Current Status

The current status of a DMA transfer can be read by software (by reading the **DMA Control/Status** register), while the DMA engine is running or after it has stopped or paused for any reason.

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 11 Interrupts

### 11.1

### Interrupt Support

The PEX 8619 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 8619 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, for assertion and de-assertion of interrupt signals), but never concurrently within the same domain. (Refer to the **PCI Command** register *Interrupt Disable* bit, offset 04h[10], and **MSI Capability** register, offset 48h, respectively.) The PEX 8619 does not convert received INTx Messages to MSI Messages.

The PEX 8619's external Interrupt output, **PEX\_INTA#**, indicates the assertion and/or de-assertion of the internally generated INTx signal:

- **Non-Hot Plug-triggered interrupts** – PEX\_INTA# assertion is controlled by the following **ECC Error Check Disable** register bits:
  - **NT mode only** – *Enable PEX\_INTA# Interrupt Output(s) for NT-Virtual Doorbell-Generated Interrupts* bit (NT Port Virtual Interface, or Port 0 if Port 0 is the NT Port, offset 1C8h[7])
  - *Enable PEX\_INTA# Interrupt Output(s) for GPIO-Generated Interrupts* (offset 1C8h[6])
  - *Enable PEX\_INTA# Interrupt Output(s) for Device-Specific Error-Triggered Interrupts* (offset 1C8h[5])

When any of these bits are Set, Device-Specific errors trigger PEX\_INTA# assertion; however, 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# assertion is controlled by the **ECC Error Check Disable** register *Enable PEX\_INTA# Interrupt Output(s) for Hot Plug or Link State Event-Triggered Interrupts* bit (offset 1C8h[4]). When this bit is Set, Hot Plug or Link State events trigger PEX\_INTA# assertion; however, an INTx Message is not generated in this case. 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 15.7, “NT Port Interrupts,”](#) for details.)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 11.1.1 Interrupt Sources or Events

The PEX 8619 internally generated interrupt/Message sources include:

- Hot Plug or Link State events:
  - Presence Detect Changed (logical OR of PRSNT# (I/O Expander 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
  - Link Bandwidth Management Status
  - Link Autonomous Bandwidth Status
- PCI Express Hot Plug events:
  - Presence Detect Changed (SerDes Receiver Detect<sup>a</sup> on Lane(s) associated with that Port)
  - Data Link Layer State Changed
- Device-Specific errors (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, unless noted otherwise):
  - Payload Link List RAM 1-Bit ECC error (offset 1C0h[1:0])
  - Payload Link List RAM 2-Bit ECC error (offset 1C0h[3:2])
  - Ingress Link List RAM 1-Bit ECC error (offset 1C0h[6, 4])
  - Ingress Link List RAM 2-Bit ECC error (offset 1C0h[7, 5])
  - Packet RAM 0 1-Bit ECC error (offset 1C0h[12, 11, 9, 8])
  - Packet RAM 0 2-Bit ECC error (offset 1C0h[18, 17, 15, 14])
  - Packet RAM 1 1-Bit ECC error (offset 1C0h[24, 23, 21, 20])
  - Packet RAM 1 2-Bit ECC error (offset 1C0h[30, 29, 27, 26])
  - Egress Completion FIFO Overflow (All Ports, offset 1CCh[0])
  - Destination Queue Linked List RAM 2-Bit ECC error (offset 1CCh[4])
  - Destination Queue Linked List RAM 1-Bit ECC error (offset 1CCh[6])
  - Source Queue Linked List RAM 1-Bit ECC error (offset 1CCh[9:8])
  - Source Queue Linked List RAM 2-Bit ECC error (offset 1CCh[11:10])
  - Retry Buffer 1-Bit ECC error (offset 1CCh[12])
  - Retry Buffer 2-Bit ECC error (offset 1CCh[13])
  - Descriptor RAM 1-Bit ECC error (offset 1CCh[17:16])
  - Descriptor RAM 2-Bit ECC error (offset 1CCh[19:18])

---

a. The SerDes Receiver Detect mechanism is comprised of the **Even/Odd Port Receiver Detect Status** register Receiver Detected on Lane x bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 200h[31:24], and Odd Ports, offset 204h[31:16]) or Hot Plug PRSNT# (from external I<sup>2</sup>C I/O Expander) input for the Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

- Header (TLP ID) RAM 2-Bit ECC error (offset 1CCh[21:20])
- Header (TLP ID) RAM 1-Bit ECC error (offset 1CCh[25:24])
- PCI Express Correctable/Uncorrectable Link errors
- *DL\_Active* state change on the NT Port Link Interface (refer to [Section 15.7, “NT Port Interrupts”](#))
- General-Purpose Input/Output (GPIO) events (upstream Port only)
- Non-Transparent (NT) Doorbell events (refer to [Section 15.7, “NT Port Interrupts”](#))

The PEX 8619 externally generated interrupt/Message sources include INTx Messages from downstream devices.

[Table 11-1](#) lists the interrupt sources.

**Table 11-1. Interrupt Sources**

Event/Error	Description
Hot Plug or Link State events	<p><b>Slot Status</b> register (Downstream Ports, offset 80h):</p> <ul style="list-style-type: none"> <li>• <i>Presence Detect Changed</i> (bit 19 is Set)</li> <li>• <i>Data Link Layer State Changed</i> (bit 24 is Set)</li> </ul>
PCI Express Hot Plug events	<p>The master control of Hot Plug interrupt is the <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Transparent Downstream Ports, offset 80h[5]). There are six sources of Hot Plug interrupt. 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>• <i>Attention Button Pressed Enable</i> (bit 0)</li> <li>• <i>Power Fault Detector Enable</i> (bit 1)</li> <li>• <i>MRL Sensor Changed Enable</i> (bit 2)</li> <li>• <i>Presence Detect Changed Enable</i> (bit 3)</li> <li>• <i>Command Completed Interrupt Enable</i> (bit 4)</li> <li>• <i>Data Link Layer State Changed Enable</i> (bit 12)</li> </ul> <p>The interrupt status of each Hot Plug source is provided by the Port’s <b>Slot Status</b> register (Transparent Downstream Ports, offset 80h).</p> <p><b>Note:</b> <i>Presence (Presence Detect State, offset 80h[22], in each Transparent downstream Port) is determined by the logical OR of:</i></p> <ul style="list-style-type: none"> <li>• <i>SerDes Receiver Detect (Even/Odd Port Receiver Detect Status register Receiver Detected on Lane x bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 200h[31:24], and Odd Ports, offset 204h[31:16])), and</i></li> <li>• <i>Hot Plug PRSNT# (from external I<sup>2</sup>C I/O Expander) input for the Port</i></li> </ul>
General-Purpose Input Interrupt events	External interrupt from any of the <b>GPIO[31:0]</b> signals that are configured as an Interrupt input in the <b>GPIO 0_15 Direction Control</b> and <b>GPIO 16_31 Direction Control</b> register <i>GPIOx Direction Control</i> bit(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 62Ch and 630h, respectively).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 11-1. Interrupt Sources (Cont.)**

Event/Error	Description
Device-Specific errors	<ul style="list-style-type: none"> <li>• Device-Specific errors indicated by the <b>Device-Specific Error Status for Egress ECC Error</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1C0h[31:0]</b>), if not masked in the <b>Device-Specific Error Mask for Egress ECC Error</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1C4h[31:0]</b>)</li> <li>• Egress Completion FIFO Overflow error indicated by the Port's <b>Error Handler 32-Bit Error Status</b> register <i>Completion FIFO Overflow Status</i> bit (offset <b>1CCh[0]</b>), if not masked by the Port's <b>Error Handler 32-Bit Error Mask</b> register <i>Completion FIFO Overflow Mask</i> bit (offset <b>1D0h[0]</b>)</li> <li>• Device-Specific errors indicated by the <b>Error Handler 32-Bit Error Status</b> register bit(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1CCh[26:4]</b>), if not masked in their corresponding <b>Error Handler 32-Bit Error Mask</b> register bit(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1D0h[26:4]</b>)</li> <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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Correctable Error Mask</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Uncorrectable Error Status</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Uncorrectable Error Mask</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[1]</b>), nor individually in the <b>Uncorrectable Error Mask</b> register (NT Port Link Interface, offset <b>FBCh</b>).</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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[2]</b>).</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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[3]</b>). 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> <li>• Device-Specific errors indicated by a <b>DMA Channel x Device-Specific Error Status</b> register (DMA of Upstream Port, offsets <b>250h</b>, <b>350h</b>, <b>450h</b>, and/or <b>550h</b>).</li> </ul>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 11-1. Interrupt Sources (Cont.)**

Event/Error	Description
<b>NT Mode Only</b>	
Device-Specific NT Port Link Interface 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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Correctable Error Mask</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Uncorrectable Error Status</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, 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 <i>Link Side Uncorrectable Error Mask</i> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[1]</b>), nor individually in the <b>Uncorrectable Error Mask</b> register (NT Port Link Interface, offset <b>FBCh</b>).</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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[2]</b>).</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 (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h[3]</b>). 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 Port Virtual Interface IRQ Set/Clear</b> register (offsets <b>C4Ch[15:0]</b> and/or <b>C50h[15:0]</b> ) bit is Set while the corresponding <b>NT Port Virtual Interface IRQ Set/Clear</b> register (offsets <b>C54h[15:0]</b> and/or <b>C58h[15:0]</b> ) bit is Cleared.
NT-Link Doorbell events	<b>NT Link Interface IRQ Set/Clear</b> register (offsets <b>C5Ch[15:0]</b> and/or <b>C60h[15:0]</b> ) bit is Set while the corresponding <b>NT Link Interface IRQ Set/Clear</b> register (offsets <b>C64h[15:0]</b> and/or <b>C68h[15:0]</b> ) bit is Cleared.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 11.1.2 Interrupt Handling

The PEX 8619 provides an Interrupt Generation module with each Port. The module reads the Request for interrupts from different sources and 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

## 11.2 INTx Emulation Support

The PEX 8619 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 *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 r2.0* provides INTA#, INTB#, INTC#, and INTD# for INTx signaling, the PEX 8619 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, 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 **PCI Status** register *Interrupt Status* bit is Set. If INTx interrupts are enabled (**PCI Command** register *Interrupt Disable* and **MSI Control** register *MSI Enable* bits, offsets **04h[10]** and **48h[16]**, respectively, are Cleared), 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#** 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 **Debug Control** register *Upstream Port and NT-Link DL\_Down Reset Propagation Disable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[20]**))
- Software Clears the corresponding *Interrupt Status* bit

## 11.2.1 INTx-Type Interrupt Message Re-Mapping and Collapsing

The upstream Port re-maps and collapses the INTx *virtual wires* received at the downstream Port, based upon the downstream Port's Device Number and Received INTx Message Requester ID Device Number, and generates a new Interrupt Message, according to the mapping defined in [Table 11-2](#).

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, due to a Hot Plug and/or PCI Express Hot Plug, Link State, GPIO, NT Port Doorbell interrupt, and/or Device-Specific error/event.

Internally generated INTx Messages always originate as type INTA Messages, because the PEX 8619 is a single-function device. Internally generated Interrupt INTA Messages from downstream Ports are re-mapped at the upstream Port to INTA, INTB, INTC, or INTD Messages, according to the mapping defined in [Table 11-2](#).

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 generates the Assert\_INTx and Deassert\_INTx Messages. The upstream Port then forwards the new Messages upstream, by way of its Link.

**Table 11-2. Downstream/Upstream Port INTx Interrupt Message Mapping**

Device Number	At Downstream Port	By Upstream Port
0, 4, 8, 12	INTA	INTA
	INTB	INTB
	INTC	INTC
	INTD	INTD
1, 5, 9, 13	INTA	INTB
	INTB	INTC
	INTC	INTD
	INTD	INTA
2, 6, 10, 14	INTA	INTC
	INTB	INTD
	INTC	INTA
	INTD	INTB
3, 7, 11, 15	INTA	INTD
	INTB	INTA
	INTC	INTB
	INTD	INTC

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 11.2.1.1 Interrupt Re-Mapping and Collapsing – NT PCI-to-PCI Bridge Mode

In NT PCI-to-PCI Bridge 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 receives an INTx Message from the NT Virtual Interface, the upstream Port 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.

## 11.3 MSI Support

One of the interrupt schemes supported by the PEX 8619 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.*

### 11.3.1 MSI Operation

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 **MSI Control** register *Multiple Message Capable* field (offset **48h[19:17]**) default value is **010b**, which indicates that the PEX 8619 requests up to four MSI Vectors (Address and Data). When the register's *Multiple Message Enable* field (offset **48h[22:20]**) is Cleared (default), only one Vector is allocated, and therefore, the PEX 8619 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 11-3 lists the supported MSI Vector types.

**Table 11-3. Supported MSI Vector Types**

Vector Type	Transparent Mode	NT Mode		DMA Mode
		Virtual Interface	Link Interface	
Power Management, or Hot Plug or Link State events	✓	✓		
Device-Specific errors	✓	✓	✓	
GPIO interrupts	✓	✓		
NT Doorbell interrupts		✓	✓	
<b>DMA (TBD)</b>				

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

System software initializes the 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 **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 **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 **MSI Control** register *Multiple Message Enable* field (offset 48h[22:20]) can be programmed to any value of 000b through 111b. When programmed to 010b through 111b, the lower three bits of Message data are changed to indicate the general type of interrupt event that occurred. (Refer to [Table 11-3](#).)

The quantity of MSI Vectors generated is dependent upon the quantity enabled, as follows:

- If **one** MSI Vector is enabled (default), all interrupt categories generate the same MSI Vector
- If **two** MSI Vectors are enabled, Device-Specific errors generate their own MSI Vector, while all other categories are combined and generate the same Vector
- If **four** MSI Vectors are enabled, each interrupt category generates its own MSI Vector

If a non-masked Interrupt event occurs before system software Sets the *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 (offset 1C8h[8]).

When the error or event that caused the interrupt is serviced, the PEX 8619 can generate a new MSI Memory Write as a result of new events. Because MSIs are edge-triggered events, four bits are provided for masking the errors (**MSI Mask** register *Interrupt Mask* bits, offset 58h[3:0]). A new MSI can be generated only after the *Interrupt Mask* bits are serviced. System software should mask these bits when the 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).

### 11.3.1.1

### NT PCI-to-PCI Bridge Mode MSIs – NT PCI-to-PCI Bridge Mode

In NT PCI-to-PCI Bridge mode, MSI TLPs are not generated if the **PCI Command** register *Bus Master Enable* bit (offset 04h[2]) is Cleared in the upstream Port, NT Port Virtual Interface, and Virtual Downstream PCI-to-PCI bridge.

### 11.3.2

### MSI Capability Registers

For details, refer to [Section 14.8, “MSI Capability Registers \(Offsets 48h – 64h\).”](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 11.4 PEX\_INTA# Interrupts

PEX\_INTA# Interrupt output is enabled when the following conditions exist:

- INTx Messages are enabled (**PCI Command** register *Interrupt Disable* bit, offset 04h[10], is Set) and MSIs are disabled (**MSI Control** register *MSI Enable* bit, offset 48h[16], is Cleared)
- PEX\_INTA# outputs are enabled for the following interrupts, when the **ECC Error Check Disable** register bit associated with that interrupt is Set:
  - **NT mode only** – *Enable PEX\_INTA# Interrupt Output(s) for NT-Virtual Doorbell-Generated Interrupts* bit (NT Port Virtual Interface, or Port 0 if Port 0 is the NT Port, offset 1C8h[7])
  - *Enable PEX\_INTA# Interrupt Output(s) for GPIO-Generated Interrupts* bit (offset 1C8h[6])
  - *Enable PEX\_INTA# Interrupt Output(s) for Device-Specific Error-Triggered Interrupts* bit (offset 1C8h[5])
  - *Enable PEX\_INTA# Interrupt Output(s) for Hot Plug or Link State Event-Triggered Interrupts* bit (offset 1C8h[4])

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# assertion

PEX\_INTA# assertion (Low) indicates that the PEX 8619 detected one or more of the events and/or errors (if not masked) listed in [Table 11-1](#).

**Note:** *PEX\_INTA# assertion and INTx messaging are mutually exclusive for a given interrupt event. When MSIs are enabled (offset 48h[16], is Set), both PEX\_INTA# and INTx are disabled for PEX 8619 internally generated interrupts. The forwarding of external INTx Messages received from a downstream Port to the upstream Port is always enabled.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 11.5 General-Purpose Input/Output

The PEX 8619 contains 32 GPIO balls and associated registers, that can be programmed to function as GPIO and/or Interrupt inputs. Default functionality is GPIO inputs; however, serial EEPROM, I<sup>2</sup>C/SMBus, and/or software can program the GPIO registers to define functionality for each I/O, to be used as GPIO, Interrupt inputs, or Serial Hot Plug PERST# outputs.

Table 11-4 lists the registers used for GPIO functionality.

**Table 11-4. Registers Used for GPIO Functionality**

Register Offset <sup>a</sup>	Register Name
62Ch	<b>GPIO 0_15 Direction Control</b>
630h	<b>GPIO 16_31 Direction Control</b>
638h	<b>GPIO 0_31 Input De-Bounce</b>
63Ch	<b>GPIO 0_15 Input Data</b>
640h	<b>GPIO 16_31 Input Data</b>
644h	<b>GPIO 0_15 Output Data</b>
648h	<b>GPIO 16_31 Output Data</b>
64Ch	<b>GPIO 0_31 Interrupt Polarity</b>
650h	<b>GPIO 0_31 Interrupt Status</b>
654h	<b>GPIO 0_31 Interrupt Mask</b>

- a. All registers listed in this table are located in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 12 Hot Plug Support

### 12.1

### Introduction

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.

### 12.2

### Hot Plug Features

The following are the PEX 8619 Hot Plug features:

- Hot insertion and removal
- All Ports are Hot Plug-capable, through I<sup>2</sup>C (Serial Hot Plug Controller on every Port)
- Serial Hot Plug Controller with I<sup>2</sup>C I/O Expander on 15 Transparent downstream Ports
- Insertion and removal of PCI Express boards, without removing system power
- Generates Power Management Event (PME) on a Hot Plug event in a sleeping system
- Electromechanical Interlock Control feature available on Serial Hot Plug-capable Ports

### 12.3

### Hot Plug Signals

The on-chip signals for Serial Hot Plug Controller support are defined in [Section 3.4.2, “Serial Hot Plug Signals.”](#) In addition to the set of on-chip Serial Hot Plug signals, the PEX 8619 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 12.6.2.](#))

### 12.4

### Hot Plug Registers

All Transparent downstream Ports include identical sets of Hot Plug registers.

The PCI Express Hot Plug Configuration, Capability, Command, Status, and Event registers are described in [Section 14.9, “PCI Express Capability Registers \(Offsets 68h – A0h\).”](#)

Device-Specific Hot Plug configuration features are programmable in register offset [1E0h](#) of each Transparent downstream Port.

### 12.5

### Hot Plug Interrupts

Refer to [Chapter 11, “Interrupts,”](#) for details.

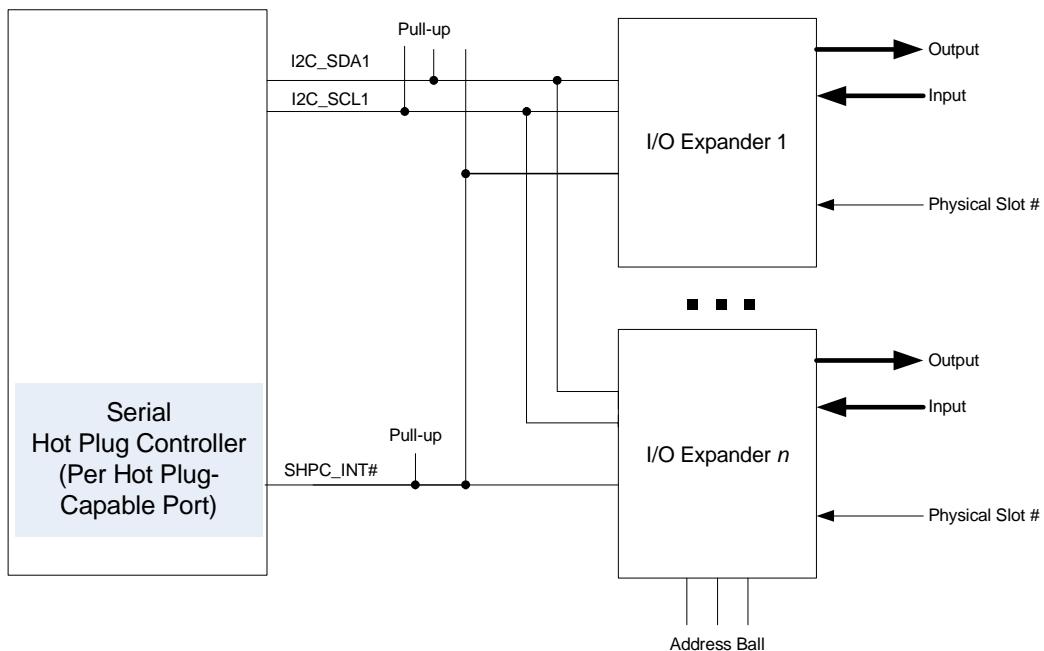
NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 12.6 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 8619 has the option of Hot Plug capability on all its Transparent downstream Ports. [Figure 12-1](#) 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 8619 reads the I/O Expander. When an I/O Expander output Port requires updating with a new value, the PEX 8619 writes to the I/O Expander through the I<sup>2</sup>C Bus.

**Figure 12-1. Serial Hot Plug Interface Diagram**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 12.6.1 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/O Expander, to assert the **PWREN** output on the I/O Expander, and thereby turn On the power. After the Write is complete and the  $T_{pepy}$  time has elapsed, the Serial Hot Plug Controller issues another Write to the I/O Expander, to assert its **RECLKEN#** output, and thereby turn On the Reference Clock (REFCLK) at the slot.

After the RECLKEN# output is asserted, the Serial Hot Plug Controller issues another Write, to de-assert the I/O Expander **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 *Command Completed* bit. 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.

## 12.6.2 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 0100\_XXXb; a Maxim MAX7311, NXP PCA9555, or TI PCA9555 is recommended. I/O Expander addresses must begin with the lowest address, and increment sequentially (corresponding to increasing PEX 8619 Port Numbers), for each device. For MAX7311 (which supports more addresses than the other 16-bit devices), the 7-bit I<sup>2</sup>C addresses can be in two ranges, 0100\_XXXb and 1010\_XXXb. All 16 I/O pins of the devices are used for one Port. A 16-bit device supports one Serial Hot Plug Port. (Refer to [Figure 12-2](#).)

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 8619 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 12-3](#).

For further details, refer to the manufacturer's data sheet for the NXP PCA9698.

The PEX 8619 can support multiple 16- or 40-bit I/O Expanders, but not both types concurrently, to provide Hot Plug services on all Transparent downstream Ports. (Refer to the **Power Management Hot Plug User Configuration** register *40-Pin I/O Expander Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [1E0h\[17\]](#))). Each I/O Expander has a 3-bit address, and thus one PEX 8619 can support up to 15 external I/O Expanders.

[Table 12-1](#) defines the external I<sup>2</sup>C I/O Expander pins, in location order.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 12-1. External I<sup>2</sup>C I/O Expander Pin Definitions, by Location**

Signal Name	Direction	Description	Location	
			16-Bit Device <sup>a</sup>	40-Bit Device <sup>b</sup>
PWRLED#	O	<b>Hot Plug Power LED Output</b>	IO0_0 or P00	IO0_0 IO2_0
ATNLED#	O	<b>Hot Plug Attention LED Output</b>	IO0_1 or P01	IO0_1 IO2_1
PWREN	O	<b>Hot Plug Power Enable Output</b>	IO0_2 or P02	IO0_2 IO2_2
RECLKEN#	O	<b>Hot Plug Reference Clock Enable Output</b>	IO0_3 or P03	IO0_3 IO2_3
PERST#	O	<b>Hot Plug Reset Output</b>	IO0_4 or P04	IO0_4 IO2_4
INTERLOCK	O	<b>Electromechanical Interlock Output Control</b> Used to physically lock the adapter or Manually operated Retention Latch (MRL) 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 (offset 80h[23]). This output can be toggled by writing 1 to the <b>Slot Control</b> register <i>Electromechanical Interlock Control</i> bit (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 (offset 7Ch[17]) is Set (default for Serial Hot Plug-capable Transparent downstream Ports).	IO0_5 or P05	IO0_5 IO2_5
GPIO	I/O	<b>General-Purpose Input/Output (GPIO)</b> Configured as input (default) or output by the <i>HPC GPIO Config</i> bit (offset 1E0h[18]), with the pin value in the <i>HPC GPIO Input/Output Value</i> bit (offset 1E0h[19]).	IO0_6 or P06	IO0_6 IO2_6
SLOT NUMBER[3:0]	I	<b>Physical Slot Number Bits 7Ch[26:23]</b> Slot Number value from I/O Expander inputs, which map to the <b>Slot Capability</b> register <i>Physical Slot Number</i> bits (offset 7Ch[26:23]). The value of the lowest four bits of the <b>Physical Slot Number</b> (offset 7Ch[22:19]) are automatically programmed to the same value as the Port Number.	{IO1_2:0, IO0_7} or {P1[2:0], P07}	{IO1_2:0, IO0_7} {IO3_2:0, IO2_7}
PRSNT#	I	<b>Hot Plug PRSNT2# Input</b>	IO1_3 or P13	IO1_3 IO3_3
MRL#	I	<b>Hot Plug Manually Operated Retention Latch Sensor Input</b>	IO1_4 or P14	IO1_4 IO3_4
BUTTON#	I	<b>Hot Plug Attention Button Input</b>	IO1_5 or P15	IO1_5 IO3_5
PWRFLT#	I	<b>Hot Plug Power Fault Input</b>	IO1_6 or P16	IO1_6 IO3_6
PWRGOOD	I	<b>Hot Plug Power Good Input</b>	IO1_7 or P17	IO1_7 IO3_7

a. Refer to [Figure 12-2](#) for pinout.

b. Refer to [Figure 12-3](#) for pinout.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 12-2. 16-Bit I<sup>2</sup>C I/O Expander Pinout**

SHPC_INT#	1	INT#	VDD	24
AD1	2	A0	SDA	23 I2C_SDA0
AD2	3	A1	SCL	22 I2C_SCL0
PWRLED#	4	IO0_0	A2	21 AD0
ATNLED#	5	IO0_1	IO1_7	20 PWRGOOD
PWREN	6	IO0_2	IO1_6	19 PWRFLT#
REFCLKEN#	7	IO0_3	PCA9555/ MAX7311 IO1_5	18 BUTTON#
PERST#	8	IO0_4	IO1_4	17 MRL#
INTERLOCK	9	IO0_5	IO1_3	16 PRSNT#
PORT_ID[0]	10	IO0_6	IO1_2	15 PORT_ID[4]
PORT_ID[1]	11	IO0_7	IO1_1	14 PORT_ID[3]
GND	12		IO1_0	13 PORT_ID[2]

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 12-3. 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	GND	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	GND	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	GND	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	GND	GND	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 12.6.3 Serial Hot Plug Port Enumeration and Assignment

Each I/O Expander has a 7-bit address; the PEX 8619 scans for sequential I<sup>2</sup>C addresses, starting with address 40h.

After **PEX\_PERST#** de-asserts and the serial EEPROM (if present) load completes, the Serial Hot Plug Controller accesses all I<sup>2</sup>C addresses to perform the I/O Expander hunt, to determine the presence of external I/O Expander devices. If the Controller receives an Acknowledge Control Packet (ACK) from the I/O Expander, it remembers the presence of an external I/O Expander device, which is assigned to one of the Transparent downstream Ports.

**Note:** *The PEX 8619 can support up to 15 downstream Ports. Every Transparent downstream Port can support Hot Plug. It is possible to support more than eight Hot Plug Ports, using 40-Bit I/O Expanders. Table 12-3 can be expanded for Serial Hot Plug Ports 8 through 15.*

To simplify the Serial Hot Plug Port enumeration and assignment, the virtual Serial Hot Plug Port definition is provided as follows, and the mapping between virtual Serial Hot Plug Port and PEX 8619 Ports is provided later.

Table 12-2 defines the virtual Serial Hot Plug Port for a 16-bit device, *such as* a Maxim MAX7311, NXP PCA9555, or TI PCA9555. Table 12-3 defines the virtual Serial Hot Plug Port for a 40-bit device, *such as* an NXP PCA9698.

The following three types of Ports are not Serial Hot Plug-capable, and therefore are not assigned a virtual Serial Hot Plug Port Number:

- Disabled Ports, due to Port configuration
- Upstream Port
- NT Port

**Table 12-2. Virtual Serial Hot Plug Port Definition for 16-Bit Device**

I <sup>2</sup> C Slave Address[2:0]	000b	001b	010b	011b	100b	101b	110b	111b
Virtual Serial Hot Plug Port #	0	1	2	3	4	5	6	7

**Table 12-3. Virtual Serial Hot Plug Port Definition for 40-Bit Device**

I <sup>2</sup> C Slave Address[2:0]	000b	000b	001b	001b	010b	010b	011b	011b	100b	100b	101b	101b	110b	110b	111b	111b
I/O Pin Range	0-15	16-31	0-15	16-31	0-15	16-31	0-15	16-31	0-15	16-31	0-15	16-31	0-15	16-31	0-15	16-31
Virtual Serial Hot Plug Port #	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 13.1 Overview

The PEX 8619 Power Management (PM) features provide the following services:

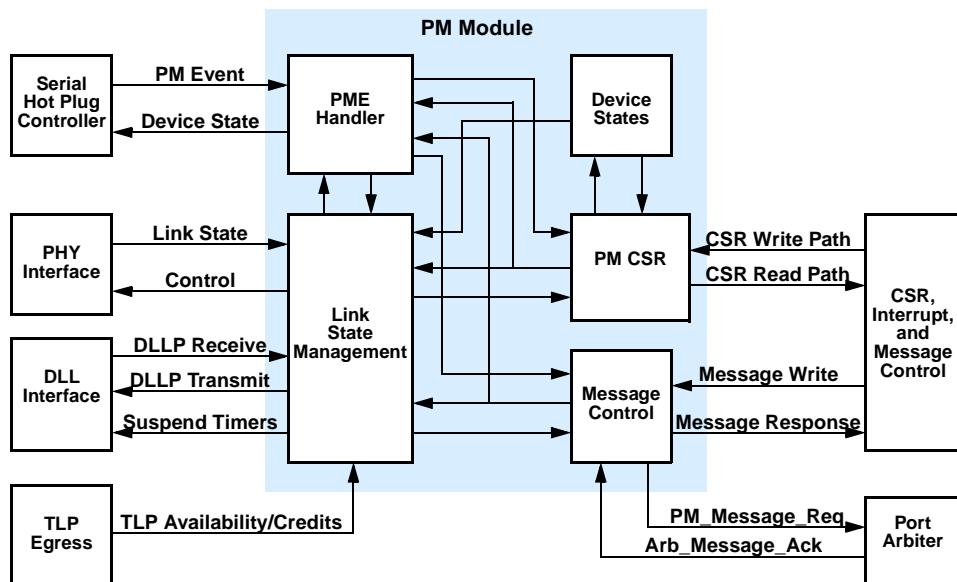
- Mechanisms to identify PM capabilities
- Ability to transition into certain PM states
- Notification of the current PM state of each Port
- Support for the option to wakeup the system upon a specific event

The PEX 8619 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 Device PM states are supported in Conventional PCI-compatible PM. Because the PEX 8619 does not support Vaux, Power Management Event (PME) generation from the D3cold Device PM 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 13-1](#) provides a functional block diagram of the PEX 8619 PM module.

**Figure 13-1. PM Module Functional Block Diagram**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 13.2 Power Management Features

- *PCI Express Base r2.0-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 *Device PM states*)
  - D0 (D0uninitialized and D0active) 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
- Supports Clock Power Management using CLK\_REQ#
- Implements Gen 2-specific Control and Status registers, and associated interrupts
- Supports ASPM L0s, ASPM L1, PCI PM L1, and L2/L3 Ready Link PM states in NT mode

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 13.3 Power Management Capability

### 13.3.1 Device Power Management States

The PEX 8619 supports the PCI Express PCI-PM D0 and D3hot Device PM states. The D1 and D2 Device PM states, which are optional in the *PCI Express Base r2.0*, are *not supported* by the PEX 8619.

The D3hot Device PM state can be entered from the D0 Device PM state, when system software programs the Port's **PCI Power Management Status and Control** register *Power State* field (offset [44h\[1:0\]](#)) to 11b. The D0uninitialized Device PM state can be entered from the D3hot Device PM state when system software Clears the Port's *Power State* field.

#### 13.3.1.1 D0 Device Power Management State

The D0 Device PM state is divided into two distinct sub-states – *uninitialized* and *active*. When power is initially applied to a PCI Express component, it defaults to the D0uninitialized Device PM state. The component remains in the D0uninitialized Device PM state until the serial EEPROM load and initial Link training completes.

A device enters the D0active Device PM 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).

#### 13.3.1.2 D3hot Device Power Management State

Once in the D3hot Device PM state, the PEX 8619 can later be transitioned into the D3cold Device PM state, by removing power from its Host component. Functions that are in the D3hot Device PM state can be transitioned, by software, to the D0uninitialized Device PM state. When in the D3hot Device PM state, Hot Plug or Link State operations cause a PME in the PEX 8619.

Only Type 0 Configuration accesses are allowed in the D3hot Device PM 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 8619 Port in the D3hot Device PM state are terminated as URs. Type 0 Configuration transactions complete successfully. When the PEX 8619 upstream Port is programmed to the D3hot Device PM state, the Port initiates Conventional PCI-PM L1 Link PM state entry.

### 13.3.2 Link Power Management States

PEX 8619 components hold their upstream Link and downstream Links in the L0 Link PM state when they are in the standard operational state (Conventional PCI-PM state is in the D0active Device PM state). ASPM defines a mechanism for components in the D0 Device PM 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 allows hardware-autonomous, dynamic Link power reduction beyond what is achievable by software-only-controlled PM. [Table 13-1](#) defines the relationship between a component's Power state and its upstream Link. [Table 13-2](#) defines the relationship between Link PM states and power-saving actions.

Conventional PCI PM, and the L1 and L2/L3 Ready Link PM states are controlled by system software programming the PEX 8619 into the D3hot Device PM state, and subsequently causing the Root Complex to broadcast the PME\_Turn\_Off Message to the downstream hierarchy.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-1. Relationship between Component Power State and Upstream Link**

Downstream Component Device PM State	Permissible Upstream Component Device PM State	Permissible Interconnect Link PM State
D0	D0	L0, L0s, L1 (optional) – ASPM.
D3hot	D0 to D3hot	L1, L2/L3 Ready.
D3cold (no Vaux)	D0 to D3cold	L3 (off). Zero power.

**Table 13-2. 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 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 13.3.3 PCI Express Power Management Support

The PEX 8619 supports PM features required in the *PCI Express Base r2.0*. Table 13-3 lists supported and non-supported features and the register bits/fields used for configuration or activation.

**Table 13-3. Supported PCI Express PM Capabilities**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
40h	<b>PCI Power Management Capability (All Ports)</b>			
	7:0	<b>Capability ID</b> Program to 01h, to indicate that the Capability structure is the PCI Power Management Capability structure.	✓	
	15:8	<b>Next Capability Pointer</b> Default 48h points to the <b>MSI Capability</b> structure.	✓	
	18:16	<b>Version</b> Default 011b indicates compliance with the <i>PCI Power Mgmt. r1.2</i> .	✓	
	19	<b>PME Clock</b> Power Management Event (PME) clock. Does not apply to PCI Express. Returns 0.		✓
	21	<b>Device-Specific Initialization</b> Default 0 indicates that Device-Specific Initialization is <i>not</i> required.	✓	
	24:22	<b>AUX Current</b> The PEX 8619 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.		✓
	25	<b>D1 Support</b> Default value of 0 indicates that the PEX 8619 does <i>not support</i> the D1 Device PM state.		✓
	26	<b>D2 Support</b> Default value of 0 indicates that the PEX 8619 does <i>not support</i> the D2 Device PM state.		✓
	31:27	<b>PME Support</b> Bits [31, 30, and 27] must be Set, to indicate that the PEX 8619 will forward PME Messages, as required by the <i>PCI Express Base r2.0</i> .	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
<b>PCI Power Management Status and Control (All Ports)</b>				
	1:0	<p><b>Power State</b>            Used to determine the current Device PM state of the Port, 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.</p>	✓	
	3	<p><b>No Soft Reset</b>            1 = Devices transitioning from the D3hot to D0 Device PM state, because of Power State commands, do not perform an internal reset</p>	✓	
	8	<p><b>PME Enable</b>            0 = Disables PME generation by the corresponding PEX 8619 Port<sup>a</sup>            1 = Enables PME generation by the corresponding PEX 8619 Port</p>	✓	
44h	12:9	<p><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 <b>Data</b> and <b>Data Scale</b> registers (fields [31:24 and 14:13], respectively).            0h = D0 power consumed            3h = D3hot power consumed            4h = D0 power dissipated            7h = D3hot power dissipated            All other encodings are <i>reserved</i>.</p>	✓	
	14:13	<p><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>Data</b> register value. The value and meaning of this field varies, depending upon which data value is selected by field [12:9] (<i>Data Select</i>).            There are four internal <b>Data Scale</b> registers (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h), per Port. For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h.</p>	✓	
	15	<p><b>PME Status</b>            0 = PME is not generated by the corresponding PEX 8619 Port<sup>a</sup>            1 = PME is being generated by the corresponding PEX 8619 Port</p>	✓	

a. Because the PEX 8619 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 00h for the **Data** and **Data Scale** registers (for all Data Selects).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
44h	<b>PCI Power Management Control/Status Bridge Extensions (All Ports)</b>			
	22	<b>B2/B3 Support</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .		✓
	23	<b>Bus Power/Clock Control Enable</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .		✓
	<b>PCI Power Management Data (All Ports)</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 <b>Data Scale</b> value returned is 0h. Selected by field [12:9] ( <i>Data Select</i> ).	✓	
	<b>Device Capability (All Ports)</b>			
	8:6	<b>Endpoint L0s Acceptable Latency</b> Because the PEX 8619 is a switch and not an endpoint, the PEX 8619 does <b>not support</b> this feature. 000b = Disables the capability		✓
6Ch	11:9	<b>Endpoint L1 Acceptable Latency</b> Because the PEX 8619 is a switch and not an endpoint, the PEX 8619 does <b>not support</b> this feature. 000b = Disables the capability		✓
	25:18	<b>Captured Slot Power Limit Value (Upstream Port and NT Port)</b> For the PEX 8619 upstream Port, 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> ). Do not change for downstream Ports.	✓	
	27:26	<b>Captured Slot Power Limit Scale (Upstream Port and NT Port)</b> For the PEX 8619 upstream Port, 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 Do not change for downstream Ports.	✓	

- a. Because the PEX 8619 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 00h for the **Data** and **Data Scale** registers (for all *Data Selects*).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
70h	<b>Device Control (All Ports)</b>			
	10	<b>AUX Power PM Enable</b>		✓
	<b>Device Status (All Ports)</b>			
	20	<b>AUX Power Detected</b>		✓
74h	<b>Link Capability (All Ports)</b>			
	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> .	✓	
	14:12	<b>L0s Exit Latency</b> Indicates the L0s Link PM state exit latency for the given PCI Express Link. Value depends upon the <b>Common Physical Layer Command/Status</b> register <i>N_FTS Value</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 238h[15:8]) value and Link speed. Exit latency is calculated, as follows: <ul style="list-style-type: none"> <li><b>2.5 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol time in 1 N_FTS) x 4 ns (1 symbol time at 2.5 GT/s)</li> <li><b>5.0 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol time in 1 N_FTS) x 2 ns (1 symbol time at 5.0 GT/s)</li> </ul> 100b = Corresponding PEX 8619 Port L0s Link PM state Exit Latency is 512 ns to less than 1 µs at 5.0 GT/s 101b = Corresponding PEX 8619 Port L0s Link PM state Exit Latency is 1 µs to less than 2 µs at 2.5 GT/s All other encodings are <i>reserved</i> .	✓	
	17:15	<b>L1 Exit Latency</b> Indicates the L1 Link PM state exit latency for the given PCI Express Link. Value depends upon the Link speed. 001b = Corresponding PEX 8619 Port L1 Link PM state Exit Latency is 1 µs to less than 2 µs at 5.0 GT/s 010b = Corresponding PEX 8619 Port L1 Link PM state Exit Latency is 2 µs to less than 4 µs at 2.5 GT/s All other encodings are <i>reserved</i> .	✓	
	18	<b>Clock Power Management Capable</b>	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
<b>Link Control (All Ports)</b>				
78h	1:0	<b>Active State Power Management (ASPM)</b> 00b = Disable <sup>c</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	✓	
	8	<b>Clock Power Management Enable</b> The PEX 8619 does <i>not support</i> removal of the Reference Clock in the L1 and L2/L3 Ready Link PM states.		✓
<b>Slot Capability (Downstream Ports; Upstream Port Always Reads 0)</b>				
7Ch	0	<b>Attention Button Present</b> <i>Reserved</i> for the upstream Port and NT Port. 0 = Attention Button is not implemented 1 = Attention Button is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	✓	
	1	<b>Power Controller Present</b> <i>Reserved</i> for the upstream Port and NT Port. 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 slot of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander. The Controller powers up the slot when the Manually operated Retention Latch (MRL) is closed and the <b>Slot Control</b> register <i>Power Controller Control</i> bit (offset 80h[10]) is Cleared. Otherwise, if bit 2 ( <i>MRL Sensor Present</i> ) is disabled (Cleared), the MRL's position has no effect on powering up the slot.	✓	
	2	<b>MRL Sensor Present</b> <i>Reserved</i> for the upstream Port and NT Port. 0 = MRL Sensor is not implemented. MRL position is “Don’t Care” for that slot. 1 = MRL Sensor is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander. The PEX 8619 senses whether the MRL is open or closed for a slot. MRL should be Low for power-on for that slot.	✓	
	3	<b>Attention Indicator Present</b> <i>Reserved</i> for the upstream Port and NT Port. 0 = Attention Indicator is not implemented 1 = Attention Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	✓	

c. The Port Receiver must be capable of entering the L0s Link PM state, regardless of whether the state is disabled.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
<b>Slot Capability (Downstream Ports; Upstream Port Always Reads 0) (Cont.)</b>				
	4	<p><b>Power Indicator Present</b>  <i>Reserved</i> for the upstream Port and NT Port.            0 = Power Indicator is not implemented            1 = Power Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I<sup>2</sup>C I/O Expander</p>	✓	
	5	<p><b>Hot Plug Surprise</b>  <i>Reserved</i> for the upstream Port and NT Port.            0 = No device in the corresponding PEX 8619 downstream Port (with an I<sup>2</sup>C I/O Expander) slot is removed from the system without prior notification            1 = Device in the corresponding PEX 8619 downstream Port slot can be removed from the system without prior notification</p>	✓	
	6	<p><b>Hot Plug Capable</b>  <i>Reserved</i> for the upstream Port and NT Port.            0 = Corresponding PEX 8619 downstream Port slot is not capable of supporting Hot Plug operations            1 = Corresponding PEX 8619 Transparent downstream Port (with an I<sup>2</sup>C I/O Expander) slot is capable of supporting Hot Plug operations</p>	✓	
7Ch	14:7	<p><b>Slot Power Limit Value</b>  <i>Reserved</i> for the upstream Port.            The maximum power supplied by the corresponding PEX 8619 downstream slot is determined by multiplying the value in this field (expressed in decimal; 25d = 19h) by the <i>Slot Power Limit Scale</i> field 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.</p>	✓	
	16:15	<p><b>Slot Power Limit Scale</b>  <i>Reserved</i> for the upstream Port.            The maximum power supplied by the corresponding PEX 8619 downstream slot is determined by multiplying the value in this field by the <i>Slot Power Limit Value</i> field 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</p>	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
80h	<b>Slot Control (Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0)</b>			
	1	<b>Power Fault Detector Enable</b> <i>Reserved</i> for the upstream Port and NT Port. 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 Device PM state ( <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 Device PM state (offset 44h[1:0], are both Set), for a <b>Power Fault Detected</b> event on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	✓	
	9:8	<b>Power Indicator Control</b> <i>Reserved</i> for the upstream Port and NT Port. Controls the Power Indicator on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot. Reads return the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port Power Indicator's current state. Writing a non-zero value triggers a Command Completed 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	✓	
	10	<b>Power Controller Control</b> <i>Reserved</i> for the upstream Port and NT Port. Controls the Power Controller on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot. 0 = Turns On the Power Controller; requires some delay to be effective 1 = Turns Off the Power Controller	✓	
	17	<b>Slot Status (Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0)</b> <b>Power Fault Detected</b> <i>Reserved</i> for the upstream Port and NT Port. 1 = Power Controller of the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot detected a Power Fault at the slot	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
<b>Power Budget Extended Capability Header (Upstream Port, and also the NT Port Link Interface)</b>				
138h	15:0	<b>PCI Express Extended Capability ID</b> Program to 0004h, as required by the <i>PCI Express Base r2.0</i> .	✓	
	19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</i> .	✓	
	31:20	<b>Next Capability Offset</b> Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	✓	
<b>Data Select (Upstream Port, and also the NT Port Link Interface)</b>				
13Ch	7:0	<b>Data Select</b> Indexes the Power Budget data reported, by way of eight upstream Port/NT Port Link Interface <b>Power Budget Data</b> registers, per 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.	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
<b>Power Budget Data (Upstream Port, and also the NT Port Link Interface)</b>				
140h	7:0	<b>Base Power</b> Eight registers per upstream Port/NT Port Link Interface. Specifies (in Watts) the base power value in the operating condition. This value must be multiplied by the field [9:8] ( <i>Data Scale</i> ) contents, to produce the actual power consumption value.	✓	
	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] ( <i>Base Power</i> ) contents with the value corresponding to the encoding returned by this field.  00b = 1.0x 01b = 0.1x 10b = 0.01x 11b = 0.001x	✓	
	12:10	<b>PM Sub-State</b> 000b = Power Management sub-state of the operating condition being described	✓	
	14:13	<b>PM State</b> Power Management state of the operating condition being described.  00b = D0 Device PM state 11b = D3 Device PM state  All other encodings are <i>reserved</i> .	✓	
	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> .	✓	
	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> .	✓	
<i>Note:</i> Eight registers, per upstream Port/NT Port Link Interface, 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 (Upstream Port, and also the NT Port Link Interface, offset <a href="#">13Ch[7:0]</a> ).				

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 13-3. Supported PCI Express PM Capabilities (Cont.)**

Register		Description	Supported	
Offset	Bit(s)		Yes	No
144h	<b>Power Budget Capability</b> (Upstream Port, and also the NT Port Link Interface)			
	0	<b>System Allocated</b> 1 = Power budget for the device is included within the system power budget	✓	
1E0h	<b>Power Management Hot Plug User Configuration (All Ports)</b>			
	0	<b>L0s Entry Idle Counter</b> Traffic idle time to meet to enter the L0s Link PM state. 0 = Idle condition must last 1 µs 1 = Idle condition must last 4 µs	✓	
	2	<b>HPC PME Turn-Off Enable</b> Functionality associated with this bit is enabled only on the downstream Ports. 1 = PME Turn-Off Message is transmitted before the Port is turned Off on a downstream Port	✓	
	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	✓	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 13.4 Power Management Tracking

*Note: NT Port Link Interface entry and exit to ASPM and Conventional PCI PM-compatible power states do not depend 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 enters 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 enters 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 exits the L1 Link PM state.
- When the upstream Port is 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. 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 transmits PM\_ENTER\_L23 DLLPs toward the Root Complex.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 13.5 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 the request of a PME.

When a PEX 8619 Transparent downstream Port is in the D3hot Device PM state, the following Hot Plug and/or PCI Express Hot Plug events cause the **PCI Power Management Status and Control** register **PME Status** bit (offset [44h\[15\]](#)) to be Set:

- For Hot Plug-capable Transparent downstream Ports
  - **Presence Detect Changed** (logical OR of PRSNT# (I/O Expander 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**
  - **Link Bandwidth Management Status**
  - **Link Autonomous Bandwidth Status**
- For non-Hot Plug-capable Transparent 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 **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 **Even/Odd Port Receiver Detect Status** register Receiver Detected on Lane x bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [200h\[31:24\]](#), and Odd Ports, offset [204h\[31:16\]](#)) or Hot Plug PRSNT# (from external I<sup>2</sup>C I/O Expander) input for the Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3



## Chapter 14 Transparent Port Registers

### 14.1 Introduction

This chapter defines the PEX 8619 Transparent Port registers. Each PEX 8619 Port has its own Configuration, Capability, Control, and Status register space. The register mapping is the same for each Port. (Refer to [Table 14-1](#).) This chapter also presents the PEX 8619 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 and downstream Ports. (Refer to [Table 14-4](#).) Other registers are defined in:

- [Chapter 16, “NT Port Virtual Interface Registers – NT Mode Only”](#)
- [Chapter 17, “NT Port Link Interface Registers – NT Mode Only”](#)
- [Chapter 18, “DMA Interface Registers – DMA Mode Only”](#)

*Note: For Chip-specific registers (those that exist only in Port 0), if Port 0 is a Legacy NT Port, then those registers exist only in the NT Port Virtual Interface.*

All PEX 8619 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-to-PCI Bridge r1.2*
- *PCI Express Base r2.0*
- *I<sup>2</sup>C Bus v2.1*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

207

## 14.2 Type 1 Port Register Map

Table 14-1 defines the Transparent mode Type 1 Port register mapping.

**Table 14-1. 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	
PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)	Capability Pointer (40h)	00h
		...
		34h
		...
		3Ch
	Next Capability Pointer (48h)      Capability ID (01h)	40h
PCI Power Management Capability Registers (Offsets 40h – 44h)		44h
	Next Capability Pointer (68h)      Capability ID (05h)	48h
		...
MSI Capability Registers (Offsets 48h – 64h)		64h
	Next Capability Pointer (A4h)      Capability ID (10h)	68h
		...
PCI Express Capability Registers (Offsets 68h – A0h)		A0h
	Next Capability Pointer (00h)      SSID/SSVID Capability ID (0Dh)	A4h
		...
Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)		FCh
Next Capability Offset (FB4h)	1h	PCI Express Extended Capability ID (0003h)
		100h
		...
Device Serial Number Extended Capability Registers (Offsets 100h – 134h)		134h
Next Capability Offset (148h)	1h	PCI Express Extended Capability ID (0004h)
		138h
		...
Power Budget Extended Capability Registers (Offsets 138h – 144h)		144h
Next Capability Offset (448h, 950h, or 520h)	1h	PCI Express Extended Capability ID (0002h)
		148h
		...
Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)		1BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 14-1. 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	
Device-Specific Registers (Offsets 1C0h – 51Ch)		
Next Capability Offset (950h or 520h)	1h	PCI Express Extended Capability ID (000Bh)
Device-Specific Registers (Offsets 1C0h – 51Ch)		
Next Capability Offset (950h)	1h	PCI Express Extended Capability ID (000Dh)
ACS Extended Capability Registers (Offsets 520h – 52Ch)		
Device-Specific Registers (Offsets 530h – F8Ch)		
Next Capability Offset 2 (000h)	1h	PCI Express Extended Capability ID 2 (000Bh)
Device-Specific Registers (Offsets 530h – F8Ch)		
<i>Factory Test Only/Reserved</i>		
Next Capability Offset (138h or 148h)	1h	PCI Express Extended Capability ID (0001h)
Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)		
<i>Reserved</i>		
		FE0h – FFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.3 Port Register Configuration and Map

The PEX 8619 Port registers are configured similarly – not all the same. Port 0 includes more Device-Specific registers than the other Ports. Port 0 also contains registers that are used to set up and control the PEX 8619, 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 each Port.

**Table 14-2** defines the Port register configuration and map.

**Table 14-2. Port Register Configuration and Map**

Register Types	Port 0	Ports 1 through 15
PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)	00h – 3Ch	00h – 3Ch
PCI Power Management Capability Registers (Offsets 40h – 44h)	40h – 44h	40h – 44h
MSI Capability Registers (Offsets 48h – 64h)	48h – 64h	48h – 64h
PCI Express Capability Registers (Offsets 68h – A0h)	68h – A0h	68h – A0h
Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)	A4h – FCh	A4h – FCh
Device Serial Number Extended Capability Registers (Offsets 100h – 134h)	100h – 134h	100h – 134h
Power Budget Extended Capability Registers (Offsets 138h – 144h)	Upstream Port 138h – 144h	
Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)	148h – 1BCh	148h – 1BCh
WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh)	When Weighted Round Robin (WRR) is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0 1A8h – 1BCh	
Device-Specific Registers (Offsets 1C0h – 51Ch)		
Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)	1C0h – 1FCh	1E0h – 1ECh, 1F8h, 1FCh
Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)	200h – 25Ch	
Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch)	260h – 26Ch	
Device-Specific Registers – Miscellaneous Control (Offset 28Ch)	28Ch	
Device-Specific Registers – I <sup>2</sup> C Slave Interface (Offsets 290h – 2C4h)	290h – 2C4h	
Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h)	2C8h – 304h	
Device-Specific Registers – I/O CAM (Offsets 308h – 340h)	308h – 340h	
Device-Specific Registers – SMBus Slave Interface (Offset 344h)	344h	
Device-Specific Registers – Address-Mapping CAM (Offsets 348h – 444h)	348h – 444h	
Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability (Offsets 448h – 51Ch)	448h – 51Ch	
ACS Extended Capability Registers (Offsets 520h – 52Ch)	520h – 52Ch	520h – 52Ch
Device-Specific Registers (Offsets 530h – F8Ch)		
Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h)	Upstream Port 530h – 540h	
Device-Specific Registers – Read Pacing (Offsets 544h – 554h)	Upstream Port 544h – 554h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 14-2. Port Register Configuration and Map (Cont.)**

<b>Register Types</b>	<b>Port 0</b>	<b>Ports 1 through 15</b>
Device-Specific Registers – Port Configuration (Offsets 574h – 628h)	574h – 628h	
Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch)	62Ch – 65Ch	
Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)	660h – 67Ch	
Device-Specific Registers – IOCAM Base and Limit Upper 16 Bits (Offsets 680h – 6BCh)	680h – 6BCh	
Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch)	6C0h – 73Ch	
Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch)	740h – 83Ch	
Device-Specific Registers – Ingress Credit Handler Port Pool (Offsets 940h – 94Ch)	940h – 94Ch	
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)	950h – 95Ch	950h – 95Ch
Device-Specific Registers – DMA Port Credit (Offsets 978h – 97Ch)	978h – 97Ch	
Device-Specific Registers – ACS Extended Capability (Offsets 980h – 9FCh)	980h – 9FCh	
Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch)	A00h – B7Ch	
Device-Specific Registers – Physical Layer (Offsets B80h – C30h)	B80h – C30h	
Device-Specific Registers – Port Configuration Header (Offsets E00h – E3Ch)	E00h – E3Ch	
Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)	E40h – EFCh	
Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch)	F00h – F3Ch	F10h
Device-Specific Registers – Error Reporting (Offsets F40h – F4Ch)	F40h – F4Ch	
Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch)	F50h – F8Ch	
Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)	FB4h – FDCh	FB4h – FDCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

Table 14-3 lists registers that are generally individual registers that support all Ports (changing the register value in one Port changes the same register in all Ports), except for the NT Port Link Interface and DMA Interface, which have their own copies (not duplicated in other Ports). Additional exceptions are also listed.

**Table 14-3. Singular Registers Shared by Multiple Ports**

Register Offset	Register Name	Comment
00h	Vendor ID and Device ID	
08h	PCI Class Code and Revision ID	NT Port Virtual Interface has its own register.
34h	Capability Pointer	
A4h	Subsystem Capability	NT Port Virtual Interface has its own register.
A8h	Subsystem Vendor ID and Subsystem ID	
100h	Device Serial Number Extended Capability Header	
104h	Serial Number (Lower DW)	
108h	Serial Number (Upper DW)	
520h	ACS Extended Capability Header	
950h	Vendor-Specific Extended Capability 2	NT Port Virtual Interface has its own register.
954h	Vendor-Specific Header 2	
958h	Hardwired Vendor ID and Hardwired Device ID	Shared by all Ports including the NT Port Link Interface.
95Ch	Hardwired Revision ID	Shared by all Ports including the NT Port Link Interface.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.4 Register Access

Each PEX 8619 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 8619 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](#)).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.4.1 PCI r3.0-Compatible Configuration Mechanism

The *PCI r3.0*-Compatible Configuration mechanism provides standard access to the PEX 8619 Ports' first 256 bytes (the bytes at offsets 00h through FFh) of the PCI Express Configuration Space. The mechanism uses PCI Type 0 and Type 1 Configuration transactions to access the PEX 8619 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 8619 decodes all Type 1 Configuration accesses received on its upstream Port, when any of the following conditions exist:

- If the Bus Number in the Configuration access is not within the upstream Port's Secondary Bus Number and Subordinate Bus Number range, the PEX 8619 upstream Port responds with an Unsupported Request (UR).
- Specified Bus Number in the Configuration access is the PEX 8619 internal virtual PCI Bus Number, the PEX 8619 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 8619 downstream Ports, the PEX 8619 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 8619 downstream Port Device Numbers, the PEX 8619 responds with a UR.
- If the specified Bus Number in the Type 1 Configuration access is not the PEX 8619 internal virtual PCI Bus Number, but is the Bus Number of one of the PEX 8619 downstream Port secondary/subordinate buses, the PEX 8619 passes the Configuration access on to the PCI Express Link attached to that PEX 8619 downstream Port.
- If the specified Bus Number is the downstream Port Secondary Bus Number, and the specified Device Number is 0, the PEX 8619 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 8619 passes along the Type 1 Configuration access, without change.

Because the mechanism is limited to the first 256 bytes of the PCI Express Configuration Space of the PEX 8619 Ports, the [PCI Express Enhanced Configuration Access Mechanism](#) or [Device-Specific Memory-Mapped Configuration Mechanism](#) must be used to access beyond Byte FFh. The PCI Express Enhanced Configuration Access mechanism can access the registers in the PCI-compatible region, as well as those in the PCI Express Extended Configuration Space.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.4.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism 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 all PEX 8619 registers.

## 14.4.3 Device-Specific Memory-Mapped Configuration Mechanism

The Device-Specific Memory-Mapped Configuration mechanism provides a method to access the Configuration registers of all Ports in a single 128-KB Memory map, as listed in [Table 14-4](#). The registers of each Port are contained within a 4-KB range. The PEX 8619 supports a maximum of 16 simultaneously active Ports.

This mechanism follows the *PCI Express Base r2.0* 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 8619 downstream Port receives a Memory Request from a downstream device targeting the PEX 8619 Configuration registers, the Port:

- Responds to a Memory Read Request with a UR
- By default:
  - Silently discards a Memory Write Request (in compliance with the *PCI Express Base r2.0*)  
–or–
  - If the Port's **ECC Error Check Disable** register *Software Force Non-Posted Request* bit (offset **1C8h[3]**) is Set, the Port responds with a UR

In Memory Requests that target PEX 8619 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 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 and/or the OS software. After the PEX 8619 upstream Port 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. (Refer to [Table 14-4](#).) Within each of these 4-KB windows, individual registers are located at the DWord offsets indicated in [Table 14-1](#).

Upstream Port **BAR0** and **BAR1** are typically enumerated at boot time, by BIOS and/or the OS software. When upstream Port **BAR0** and **BAR1** are written (by serial EEPROM, I<sup>2</sup>C Slave interface, and/or software), the PEX 8619 automatically copies the values into the Port 0 **BAR0** and **BAR1 Shadow** registers. (Refer to [Table 14-38](#) for register mapping.) The particular registers used within this block depend upon which Port is the upstream Port.

If upstream Port **BAR0** and **BAR1** are enumerated by serial EEPROM, rather than by BIOS/OS, the serial EEPROM must be programmed to also load the same values to the corresponding **BAR0** and **BAR1 Shadow** registers.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-4. Register Offsets from Upstream Port BAR0/1 Base Address**

Port Number	Internal Register 4-KB Memory Space Range	Location Range
Port 0	0000h to 0FFFh	0 to 4 KB
Port 1	1000h to 1FFFh	4 to 8 KB
Port 2	2000h to 2FFFh	8 to 12 KB
Port 3	3000h to 3FFFh	12 to 16 KB
Port 4	4000h to 4FFFh	16 to 20 KB
Port 5	5000h to 5FFFh	20 to 24 KB
Port 6	6000h to 6FFFh	24 to 28 KB
Port 7	7000h to 7FFFh	28 to 32 KB
Port 8	8000h to 8FFFh	32 to 36 KB
Port 9	9000h to 9FFFh	36 to 40 KB
Port 10	A000h to AFFFh	40 to 44 KB
Port 11	B000h to BFFFh	44 to 48 KB
Port 12	C000h to CFFFh	48 to 52 KB
Port 13	D000h to DFFFh	52 to 56 KB
Port 14	E000h to EFFFh	56 to 60 KB
Port 15	F000h to FFFFh	60 to 64 KB

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.5 Register Descriptions

The remainder of this chapter details the PEX 8619 registers, including:

- Bit/field names
- Description of register functions for the PEX 8619 upstream Port and downstream Ports
- Type (*such as RW or HwInit; refer to Table 14-5 for Type descriptions*)
- Whether the power-on/reset value can be modified, by way of the PEX 8619 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

**Table 14-5. Register Types, Grouped by User Accessibility**

Type	Description
HwInit	<b>Hardware-Initialized</b> Refers to the PEX 8619 Hardware-Initialization mechanism or PEX 8619 Serial EEPROM 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 8619 Hardware-Initialization mechanism or PEX 8619 serial EEPROM and/or I <sup>2</sup> C 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> <b>Reserved</b> for future RW implementations. Registers are RO and must return 0 when read. Software must preserve value read for Writes to bits.
RsvdZ	<b>Reserved and Zero</b> <b>Reserved</b> for future RW1C implementations. Registers are RO and must return 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, Doorbell interrupts)</b> Indicates 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, Doorbell interrupts)</b> Indicates 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 0; however, software is allowed to write this register.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.6 PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)

This section details the PCI-Compatible Type 1 Configuration Header registers. [Table 14-6](#) defines the register map.

**Table 14-6. PCI-Compatible Type 1 Configuration Header 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	
	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

**Register 14-1. 00h PCI Configuration ID (All Ports)**

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 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-2. 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 8619 ignores I/O Space accesses on the corresponding Port's primary interface 1 = PEX 8619 responds to I/O Space accesses on the corresponding Port's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on the corresponding Port's primary interface 1 = PEX 8619 responds to Memory Space accesses on the corresponding Port's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream.  0 = PEX 8619 handles Memory and I/O Requests received on the corresponding Port downstream/secondary interface as Unsupported Requests (URs); for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status. Because MSI Messages are in-band Memory Writes, disables MSI Messages as well. 1 = PEX 8619 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 r2.0</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 r2.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</i> .	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-2. 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 r2.0</i> .	RsvdP	No	0
10	<b>Interrupt Disable</b> 0 = Corresponding PEX 8619 Port is enabled to generate INTx Interrupt Messages and assert <b>PEX_INTA#</b> output 1 = Corresponding PEX 8619 Port is prevented from generating INTx Interrupt Messages and asserting <b>PEX_INTA#</b> output	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-2. 04h PCI Command/Status  
(All Ports) (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 Message is pending 1 = INTx Interrupt Message is pending internally to the corresponding PEX 8619 Port –or– <b>PEX_INTA#</b> (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 r2.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r2.0</i> .	RsvdP	No	0
22	<b>Reserved</b>	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 r2.0</i> .	RsvdP	No	0
24	<b>Master Data Parity Error Detected</b> If bit <b>6 (Parity Error Response Enable)</b> is Set, the corresponding PEX 8619 Port Sets this bit when the Port: <ul style="list-style-type: none"> <li>Forwards the poisoned Transaction Layer Packet (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 8619 never Sets this bit. This error is natively reported by the <b>Uncorrectable Error Status</b> register <b>Poisoned TLP Status</b> 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 r2.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> The upstream Port Sets this bit if one of the following conditions exist: <ul style="list-style-type: none"> <li>Upstream Port receives a Memory Request targeting a PEX 8619 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 8619 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 <b>Completer Abort Status</b> bit (offset FB8h[15]), which is mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-2. 04h PCI Command/Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
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 corresponding PEX 8619 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 = Corresponding 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 14-3. 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 8619. The PEX 8619 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>				
15:8	<b>Register-Level Programming Interface</b> The PEX 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-4. 0Ch Miscellaneous Control  
(All Ports)**

Bit(s)	Description	Ports	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 8619 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 r2.0</i> .	RsvdP	No		00h
<b>PCI Header Type</b>					
22:16	<b>Configuration Layout Type</b> The corresponding PEX 8619 Port 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	Upstream	RO	No	1
		Downstream	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 -- kotlin-novator | vtuz vova**

**Register 14-5. 10h Base Address 0  
(Upstream Port)**

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 8619 Configuration registers into Memory space <i>Note: The upstream Port is hardwired to 0.</i>	Upstream	RO	No	0
	<b>Reserved</b>	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
	<b>Reserved</b>	Downstream	RsvdP	No	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8619 Configuration registers into Non-Prefetchable Memory space <i>Note: The upstream Port is hardwired to 0.</i>	Upstream	RO	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
16:4	<b>Reserved</b>		RsvdP	No	0-0h
31:17	<b>Base Address 0</b> Base Address ( <b>BAR0</b> ) for the Device-Specific Memory-Mapped Configuration mechanism.	Upstream	RW	Yes	0-0h
	<b>Reserved</b>	Downstream	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-6. 14h Base Address 1  
(Upstream Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1</b> RO when the <b>Base Address 0 (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).	Upstream	RO	Yes	0000_0000h
	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.	Downstream	RW	Yes	0000_0000h

**Register 14-7. 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. Set 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. Set 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. Set 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 r2.0</i> .	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### Register 14-8. 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
<b>Note:</b> If ISA Addressing mode is enabled ( <b>PCI Command</b> register <b>I/O Access Enable</b> bit, offset 04h[0], is Set), the PEX 8619 Port forwards I/O transactions from its primary interface to its 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.				
Conversely, the PEX 8619 Port forwards I/O transactions from its secondary interface to its primary interface (upstream) if an I/O address is outside this Address range. If the PEX 8619 Port does not implement an I/O Address range, the Port forwards all I/O transactions on its secondary interface upstream, to its primary interface.				
<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 <i>reserved</i> .	RO	Yes	1h
7:4	<b>I/O_BAR[15:12]</b> I/O Base Address[15:12]. The PEX 8619 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 corresponding PEX 8619 Port I/O Base Address[15:12]. The PEX 8619 assumes I/O Base Address[11:0]=000h. For 16-bit I/O addressing, the PEX 8619 assumes Address[31:16]=0000h. For 32-bit addressing, the PEX 8619 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 <i>reserved</i> .	RO	Yes	1h
15:12	<b>I/O_Limit[15:12]</b> I/O Limit Address[15:12]. The PEX 8619 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 corresponding PEX 8619 Port I/O Limit Address[15:12]. The PEX 8619 assumes Address bits [11:0] of the I/O Limit Address are FFFh. For 16-bit I/O addressing, the PEX 8619 decodes Address bits [15:0] and assumes Address bits [31:16] of the I/O Limit Address are 0000h. For 32-bit addressing, the PEX 8619 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). If the I/O Limit Address is less than the I/O Base Address, the PEX 8619 does not forward I/O transactions from the corresponding Port primary/upstream bus to its secondary/downstream bus. However, the PEX 8619 forwards all I/O transactions from the secondary bus of the corresponding Port to its primary bus.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-8. 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	<b>Reserved</b>	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	<b>Reserved</b>	RsvdP	No	0
23	<b>Fast Back-to-Back Transactions Capable</b> <b>Reserved</b> 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 corresponding PEX 8619 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 8619 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 r2.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> Cleared, as required by the <i>PCI Express Base r2.0</i> .	RsvdP	No	0
28	<b>Received Target Abort</b> Cleared, as required by the <i>PCI Express Base r2.0</i> , because the PEX 8619 never initiates a Request itself.	RsvdP	No	0
29	<b>Received Master Abort</b> Cleared, as required by the <i>PCI Express Base r2.0</i> , because the PEX 8619 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, regardless of the <b>Bridge Control</b> register <i>Parity Error Response Enable</i> bit (offset 3Ch[16]) state	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### Register 14-9. 20h Memory Base and Limit (All Ports)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The PEX 8619 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).				
Conversely, the PEX 8619 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).				
<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 corresponding PEX 8619 Port Non-Prefetchable Memory Base Address[31:20]. The PEX 8619 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 corresponding PEX 8619 Port Non-Prefetchable Memory Limit Address[31:20]. The PEX 8619 assumes Memory Limit Address[19:0]=F_FFFFh.	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-10. 24h Prefetchable Memory Base and Limit  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The PEX 8619 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 28h + 24h[15:0]) and <b>Prefetchable Memory Limit</b> (offsets 2Ch + 24h[31:16]) registers (when the Base is less than or equal to the Limit).				
Conversely, the PEX 8619 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 20h)).				
<b>Prefetchable Memory Base</b>				
0	<b>Prefetchable Memory Base Capability</b> 0 = Corresponding PEX 8619 Port supports 32-bit Prefetchable Memory Addressing 1 = Corresponding PEX 8619 Port defaults to 64-bit Prefetchable Memory Addressing support, as required by the <i>PCI Express Base r2.0</i> <i>Note:</i> 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 corresponding PEX 8619 Port Prefetchable Memory Base Address[31:20]. The PEX 8619 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 = Corresponding PEX 8619 Port supports 32-bit Prefetchable Memory Addressing 1 = Corresponding PEX 8619 Port defaults to 64-bit Prefetchable Memory Addressing support, as required by the <i>PCI Express Base r2.0</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 corresponding PEX 8619 Port Prefetchable Memory Limit Address[31:20]. The PEX 8619 assumes Prefetchable Memory Limit Address[19:0]=F_FFFFh.	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-11. 28h Prefetchable Memory Upper Base Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	PBUP[63:32] Prefetchable Memory Base Address[63:32]. The PEX 8619 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 0000_0000h.	Offset 24h[0]=0	RO	No 0000_0000h

**Register 14-12. 2Ch Prefetchable Memory Upper Limit Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	PLIMUP[63:32] Prefetchable Memory Limit Address[63:32]. The PEX 8619 uses this register for Prefetchable Memory Upper Limit Address[63:32].	Offset 24h[16]=1	RW	Yes 0000_0000h
	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 0000_0000h.  <i>Note:</i> The serial EEPROM must not write a non-zero value into this register when the RO attribute is Set for this register.	Offset 24h[16]=0	RO	No 0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-13. 30h I/O Upper Base and Limit Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
15:0	<b>I/O Base Upper 16 Bits</b> The PEX 8619 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 0000h.	Offset 1Ch[3:0]=1h	RW	Yes	0000h
		Offset 1Ch[3:0]=0h	RO	No	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> The PEX 8619 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 0000h.  <i>Note:</i> The serial EEPROM must not write a non-zero value into this register when the RO attribute is Set for this register.	Offset 1Ch[11:8]=1h	RW	Yes	0000h
		Offset 1Ch[11:8]=0h	RO	No	0000h

**Register 14-14. 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	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 14-15. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-16. 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 Line</b>				
7:0	<b>Interrupt Line Routing Value</b> The PEX 8619 does <i>not</i> use this register; however, the register is included for operating system and device driver use.	RW	Yes	00h
<b>PCI Interrupt Pin</b>				
15:8	<b>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 8619. 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 <b>Secondary Status</b> register <i>Master Data Parity Error</i> bit (offset 1Ch[24]) 1 = Enables the <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 <b>PCI Command</b> register <i>SERR# Enable</i> bit (offset 04h[8]) is Set, enables the <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 8619's response to ISA I/O addresses enabled by the <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). The default state of this bit after reset is 0. 0 = If ISA Addressing mode is enabled ( <b>PCI Command</b> register <i>I/O Access Enable</i> bit, offset 04h[0], is Set), the PEX 8619 Port forwards I/O Requests within the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers. 1 = PEX 8619 blocks forwarding from the primary to secondary interface, of I/O transactions addressing the last 768 bytes in each 1-KB block of the Port's I/O Address range. In the opposite direction (secondary to primary), if I/O Addressing mode is enabled, the PEX 8619 Port forwards I/O transactions that address the last 768 bytes in each 1-KB block of the Port's I/O Address range. <i>Note:</i> Refer also to the <b>Ingress Control</b> register <i>Disable VGA BIOS Memory Access Decoding</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 660h[28]).	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-16. 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> Modifies the bridge response to VGA-compatible addresses. 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 <b>PCI Command</b> register <b>Memory Access Enable</b> and <b>I/O Access Enable</b> 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 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><i>Notes:</i> When Set in an egress Port, the Port is configured as a non-Cut-Thru path. (Refer to <i>Section 2.1.3.2, “Cut-Thru,”</i> for further details.)</p> <p>Refer also to the <b>Ingress Control</b> register <b>Disable VGA BIOS Memory Access Decoding</b> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 660h[28]).</p> <p><b>Conventional PCI VGA support</b> – To avoid potential I/O address conflicts, if the VGA Enable bit is Set in the upstream Port and a downstream Port, Set the <b>PCI Command</b> register I/O Access Enable 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 Set above 1_0000h.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-16. 3Ch Bridge Control and PCI Interrupt Signal  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<p><b>VGA 16-Bit Decode Enable</b> Used only when bit 19 (<a href="#">VGA Enable</a>) 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</p> <p><i>Note:</i> Refer also to the <a href="#">Ingress Control</a> register <a href="#">Disable VGA BIOS Memory Access Decoding</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">660h</a>[28]).</p>	RW	Yes	0
21	<p><b>Master Abort Mode</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
22	<p><b>Secondary Bus Reset</b> 1 = Causes a Hot Reset on the corresponding PEX 8619 Port downstream Link</p>	RW	Yes	0
23	<p><b>Fast Back-to-Back Transactions Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
24	<p><b>Primary Discard Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
25	<p><b>Secondary Discard Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
26	<p><b>Discard Timer Status</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
27	<p><b>Discard Timer SERR# Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</i>.</p>	RsvdP	No	0
31:28	<b>Reserved</b>	RsvdP	No	0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.7 PCI Power Management Capability Registers (Offsets 40h – 44h)

This section details the PCI Power Management Capability registers. Table 14-7 defines the register map.

**Table 14-7. PCI Power Management 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		
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 14-17. 40h PCI Power Management Capability (All Ports)**

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	48h
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 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 8619 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 8619 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8619 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 8619 will forward PME Messages, as required by the <i>PCI Express Base r2.0</i> .	RO	Yes	19h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-18. 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	<p><b>Power State</b> Used to determine the current Device PM state of the Port, and to program the Port into a new Device PM state.</p> <p>00b = D0 01b = D1 – <i>Not supported</i> 10b = D2 – <i>Not supported</i> 11b = D3hot</p> <p>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.</p>	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<p><b>No Soft Reset</b> 1 = Devices transitioning from the D3hot to D0 Device PM state, because of Power State commands, do not perform an internal reset</p>	RO	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<p><b>PME Enable</b> 0 = Disables PME generation by the corresponding PEX 8619 Port<sup>a</sup> 1 = Enables PME generation by the corresponding PEX 8619 Port</p>	RWS	No	0
12:9	<p><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 <b>Data</b> and <b>Data Scale</b> registers (fields [31:24 and 14:13], respectively).</p> <p>0h = D0 power consumed 3h = D3hot power consumed 4h = D0 power dissipated 7h = D3hot power dissipated All other encodings are <i>reserved</i>.</p>	RO	Yes	0h
14:13	<p><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>Data</b> register value. The value and meaning of this field varies, depending upon which data value is selected by field [12:9] (<i>Data Select</i>). There are four internal <b>Data Scale</b> registers (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h), per Port. For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h.</p>	RO	Yes	00b
15	<p><b>PME Status</b> 0 = PME is not generated by the corresponding PEX 8619 Port<sup>a</sup> 1 = PME is being generated by the corresponding PEX 8619 Port</p>	RW1CS	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-18. 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	<b>Reserved</b>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <b>Reserved</b> 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 <b>Data Scale</b> value returned is 0h. Selected by field [12:9] ( <i>Data Select</i> ).	RO	Yes	00h

- a. Because the PEX 8619 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 00h for the **Data** and **Data Scale** registers (for all Data Selects).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.8 MSI Capability Registers (Offsets 48h – 64h)

This section details the Message Signaled Interrupt (MSI) Capability registers. [Table 14-8](#) defines the register map.

**Table 14-8. MSI Capability Register Map  
(All 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	
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 14-19. 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 corresponding Port are disabled 1 = MSIs for the corresponding Port are enabled, and INTx Interrupt Messages and <b>PEX_INTA#</b> output assertion are disabled	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = PEX 8619 Port can request only one Vector 001b = PEX 8619 Port can request two Vectors 010b through 111b = PEX 8619 Port can request four Vectors	RO	Yes	010b
22:20	<b>Multiple Message Enable</b> 000b = PEX 8619 Port is allocated one Vector, by default 001b = PEX 8619 Port is allocated two Vectors 010b through 111b = PEX 8619 Port is allocated four Vectors <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 8619 is capable of generating MSI 32-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the Message address) 1 = PEX 8619 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 8619 does not have Per Vector Masking capability 1 = PEX 8619 has Per Vector Masking capability	RO	Yes	1
31:25	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-20. 4Ch MSI Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b>	RsvdP	No	00b
31:2	<b>Message Address</b> <i>Note:</i> Refer to register offset 50h for MSI Upper Address, if offset 48h[23] is Set (default).	RW	Yes	0-0h

**Register 14-21. 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 <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:</i> Refer to register offset 4Ch for MSI Address.	RW	Yes	0000_0000h

**Register 14-22. 54h MSI Data  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The offset for this register changes from 54h, to 50h, when the <b>MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Cleared.				
15:0	<b>Message Data</b> MSI Write transaction TLP payload.	RW	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-23. 58h MSI Mask  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
The interrupt sources in a PEX 8619 Port are grouped into four categories – Power Management/Hot Plug or Link State events, Device-Specific errors, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts.					
The quantity of allocated MSI Vectors is determined by the <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). When the quantity of MSI Vectors that can be requested is:					
<ul style="list-style-type: none"> <li>• <b>Four</b> – Each interrupt category generates its own MSI Vector</li> <li>• <b>Two</b> – Device-Specific errors generate their own MSI Vector, while all other categories are combined and generate the same Vector</li> <li>• <b>One</b> – All interrupt categories generate the same MSI Vector</li> </ul>					
NT-Virtual Doorbell interrupts are generated only on the NT Port. The Type is the same, regardless of whether it is for a Transparent or NT Port.					
<i>Notes:</i> 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.					
The bits in this register can be used to mask their respective <b>MSI Status</b> register bits (offset 5Ch).					
0	<b>MSI Mask for Power Management, or Hot Plug or Link State, Events</b> MSI mask for Power Management event- or Hot Plug or Link State event-generated interrupts.	Offset 48h[22:20]≥010b	RW	Yes	0
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 Errors</b> MSI mask for Device-Specific error-generated interrupts.	Offset 48h[22:20]≥001b	RW	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0
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> <i>This bit is valid only in NT mode.</i> Refer to the 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### Register 14-24. 5Ch MSI Status (All Ports)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
	<p>The interrupt sources in a PEX 8619 Port are grouped into four categories – Power Management/Hot Plug or Link State events, Device-Specific errors, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts.</p> <p>The quantity of allocated MSI Vectors is determined by the <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). When the quantity of MSI Vectors that can be requested is:</p> <ul style="list-style-type: none"> <li><b>Four</b> – Each interrupt generates its own MSI Vector</li> <li><b>Two</b> – Device-Specific errors generate their own MSI Vector, while all other categories are combined and generate the same Vector</li> <li><b>One</b> – All interrupt categories generate the same MSI Vector</li> </ul> <p>NT-Virtual Doorbell interrupts are generated only on the NT Port. The Type is the same, regardless of whether it is for a Transparent or NT Port.</p> <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	<p><b>MSI Pending Status for Power Management, or Hot Plug or Link State, Events</b> MSI pending status for Power Management event- or Hot Plug or Link State event-generated interrupts.</p> <p><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.</p>	Offset 48h[22:20] $\geq$ 010b  Offset 48h[22:20] $\leq$ 001b	RO  RO	Yes  Yes	0  0
1	<p><b>MSI Pending Status for Device-Specific Errors</b> MSI pending status for Device-Specific error-generated interrupts.</p> <p><b>Reserved</b></p>	Offset 48h[22:20] $\geq$ 001b  Offset 48h[22:20]=000b	RO  RsvdP	Yes  No	0  0
2	<p><b>MSI Pending Status for GPIO-Generated Interrupts</b></p> <p><b>Reserved</b></p>	Offset 48h[22:20] $\geq$ 010b  Offset 48h[22:20] $\leq$ 001b	RO  RsvdP	Yes  No	0  0
3	<p><b>MSI Pending Status for NT-Virtual Doorbell-Generated Interrupts</b> <i>This bit is valid only in NT mode.</i> Refer to the NT Port registers located at offsets C4Ch through C58h.</p> <p><b>Reserved</b></p>	Offset 48h[22:20] $\geq$ 010b  Offset 48h[22:20] $\leq$ 001b	RO  RsvdP	Yes  No	0  0
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 14.9 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 14-9](#) defines the register map.

**Table 14-9. 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)
Device Capability		
Device Status	<i>Not Supported/Reserved</i>	Device Control
Link Capability		
Link Status		Link Control
<i>Reserved</i> (Upstream) Slot Capability (Downstream)		
<i>Reserved</i> (Upstream)		
Slot Status (Downstream)		Slot Control (Downstream)
<i>Reserved</i>		84h – 88h
Device Capability 2		8Ch
Device Status 2 ( <i>Reserved</i> )		90h
<i>Reserved</i>		94h
Link Status 2		Link Control 2
<i>Reserved</i>		9Ch – A0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-25. 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 r2.0</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 8619 Ports program this field to 2h, as required by the <i>PCI Express Base r2.0</i> .		RO	Yes	2h
23:20	<b>Device/Port Type</b> Set at reset, as required by the <i>PCI Express Base r2.0</i> .	Upstream	RO	Yes	5h
		Downstream	RO	Yes	6h
24	<b>Slot Implemented</b> 0 = Disables or connects to the 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 8619 serial EEPROM register Initialization capability and/or and/or I<sup>2</sup>C can be used to Clear this bit, indicating that the corresponding PEX 8619 downstream Port connects to an integrated component or is disabled.</i>	Downstream	RO	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

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-26. 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> Maximum Payload Size Port limitations are as follows: If there are no x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 2,048 bytes. If there are x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 512 bytes. 000b = PEX 8619 Port supports a 128-byte maximum payload 001b = PEX 8619 Port supports a 256-byte maximum payload 010b = PEX 8619 Port supports a 512-byte maximum payload 011b = PEX 8619 Port supports a 1,024-byte maximum payload 100b = PEX 8619 Port supports a 2,048-byte maximum payload No other encodings are supported.		HwInit	Yes	100b
4:3	<b>Phantom Functions Supported</b> <i>Not supported</i>		RO	Yes	00b
5	<b>Extended Tag Field Supported</b> 0 = Maximum Tag field is 5 bits 1 = Maximum Tag field is 8 bits		RO	Yes	0
8:6	<b>Endpoint L0s Acceptable Latency</b> <i>Not supported</i> Because the PEX 8619 is a switch and not an endpoint, the PEX 8619 does <i>not support</i> this feature. 000b = Disables the capability		RO	Yes	000b
11:9	<b>Endpoint L1 Acceptable Latency</b> <i>Not supported</i> Because the PEX 8619 is a switch and not an endpoint, the PEX 8619 does <i>not support</i> this feature. 000b = Disables the capability		RO	Yes	000b
14:12	<b>Reserved</b> , as required by the PCI Express Base r2.0.		RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>		RO	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-26. 6Ch Device Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	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 PEX 8619 upstream Port, 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
	<b>Not valid</b>	Downstream	RsvdP	No	00h
27:26	<b>Captured Slot Power Limit Scale</b> For the PEX 8619 upstream Port, 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
	<b>Not valid</b>	Downstream	RsvdP	No	00b
31:28	<b>Reserved</b>		RsvdP	No	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-27. 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 corresponding PEX 8619 Port to report Correctable errors	RW	Yes	0
1	<b>Non-Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the corresponding PEX 8619 Port to report Non-Fatal errors	RW	Yes	0
2	<b>Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the corresponding PEX 8619 Port to report Fatal errors	RW	Yes	0
3	<b>Unsupported Request Reporting Enable</b> 0 = Disables 1 = Enables the corresponding PEX 8619 Port to report UR errors	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> <i>Not supported</i>	RsvdP	No	0
7:5	<b>Maximum Payload Size</b> Software can change this field to configure the PEX 8619 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 = PEX 8619 Port supports a 128-byte maximum payload 001b = PEX 8619 Port supports a 256-byte maximum payload 010b = PEX 8619 Port supports a 512-byte maximum payload 011b = PEX 8619 Port supports a 1,024-byte maximum payload 100b = PEX 8619 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>	RsvdP	No	0
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 r2.0</i> .	RsvdP	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-27. 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> Set when the corresponding Port detects a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state.  0 = Corresponding PEX 8619 Port did not detect a Correctable error 1 = Corresponding PEX 8619 Port detected a Correctable error	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> Set when the corresponding Port detects a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state.  0 = Corresponding PEX 8619 Port did not detect a Non-Fatal error 1 = Corresponding PEX 8619 Port detected a Non-Fatal error	RW1C	Yes	0
18	<b>Fatal Error Detected</b> Set when the corresponding Port detects a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state.  0 = Corresponding PEX 8619 Port did not detect a Fatal error 1 = Corresponding PEX 8619 Port detected a Fatal error	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> Set when the corresponding Port detects a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state.  0 = Corresponding PEX 8619 Port did not detect a UR 1 = Corresponding PEX 8619 Port detected a UR	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 r2.0</i> .	RsvdP	No	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-28. 74h Link Capability  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Supported Link Speeds</b> Indicates the Port's supported Link speed. 0001b = 2.5 GT/s Link speed is supported 0010b = 5.0 GT/s and 2.5 GT/s Link speeds are supported All other encodings are <i>reserved</i>.</p>		RO	Yes	0010b (STRAP_RESERVED17#=H)  0001b (STRAP_RESERVED17#=L)
9:4	<p><b>Maximum Link Width</b> The PEX 8619 maximum Link width is x8 = 00_1000b. Valid widths are x1, x4, or x8. Actual maximum Link width is Set by the <a href="#">STRAP_PORTCFG[3:0]</a> inputs.</p> <p>00_0000b = <i>Reserved</i> 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8  All other encodings are <i>not supported</i>.</p>		ROS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <a href="#">Port Configuration</a> register <a href="#">Port Configuration</a> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
11:10	<p><b>Active State Power Management (ASPM) Support</b> Active State Link PM support. Indicates the level of ASPM supported by the Port.</p> <p>01b = L0s Link PM state entry is supported 11b = L0s and L1 Link PM states are supported  All other encodings are <i>reserved</i>.</p>		RO	Yes	11b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-28. 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> Indicates the L0s Link PM state exit latency for the given PCI Express Link. Value depends upon the <b>Common Physical Layer Command/Status</b> register <i>N_FTS Value</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 238h[15:8]) value and Link speed. Exit latency is calculated, as follows:</p> <ul style="list-style-type: none"> <li>• <b>2.5 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 N_FTS) x 4 ns (1 symbol time at 2.5 GT/s)</li> <li>• <b>5.0 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 N_FTS) x 2 ns (1 symbol time at 5.0 GT/s)</li> </ul> <p>100b = Corresponding PEX 8619 Port L0s Link PM state Exit Latency is 512 ns to less than 1 µs at 5.0 GT/s 101b = Corresponding PEX 8619 Port L0s Link PM state Exit Latency is 1 µs to less than 2 µs at 2.5 GT/s All other encodings are <i>reserved</i>.</p>		RO	No	100b (5.0 GT/s)  101b (2.5 GT/s)
17:15	<p><b>L1 Exit Latency</b> Indicates the L1 Link PM state exit latency for the given PCI Express Link. Value depends upon the Link speed.</p> <p>001b = Corresponding PEX 8619 Port L1 Link PM state Exit Latency is 1 µs to less than 2 µs at 5.0 GT/s 010b = Corresponding PEX 8619 Port L1 Link PM state Exit Latency is 2 µs to less than 4 µs at 2.5 GT/s All other encodings are <i>reserved</i>.</p>		RO	Yes	001b (5.0 GT/s)  010b (2.5 GT/s)
18	<b>Clock Power Management Capable</b>		RO	Yes	0
19	<p><b>Reserved</b> This bit must be hardwired to 0, for the upstream Port and components that do not support this optional capability.</p>	Upstream	RsvdP	No	0
	<p><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 <b>Uncorrectable Error Status</b> register <i>Surprise Down Error Status</i> bit (offset FB8h[5]) is disabled.</p> <p><i>Note:</i> If this bit is Set and later Cleared at runtime (such as by I<sup>2</sup>C), it must be Cleared while the Link is up; otherwise, if the Link is down when this bit is Cleared, a subsequent Surprise Down error event is not masked.</p>	Downstream	RO	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-28. 74h Link Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>  <b>Data Link Layer Link Active Reporting Capable</b> Valid for downstream Ports only.	Upstream	RsvdP	No	0
21	<b>Reserved</b> Hardwired to 0, as required by the <i>PCI Express Base r2.0</i> .  <b>Link Bandwidth Notification Capability</b> 1 = Indicates support for the Link Bandwidth Notification status and interrupt mechanisms	Downstream	RO	Yes	1
23:22	<b>Reserved</b>	Upstream	RsvdP	No	00b
31:24	<b>Port Number</b> The Port Number is defined by the <b>STRAP_PORTCFG[3:0]</b> input Strapping options. (Refer to <b>Table 14-10</b> . Encodings not listed are <i>reserved</i> .) Available Port Numbers are 0 through 15.	Downstream	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>574h[3:0]</b> )

**Table 14-10. Port Configurations**

STRAP_PORTCFG [3:0]	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15
(default) 0h	x1	x1	x1	x1	x1	x1										
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	x1
2h	x4	x4	x1	x1							x1	x1	x1	x1	x1	x1
3h	x4	x4	x4	x1								x1		x1	x1	x1
4h	x4	x4	x4	x4												
5h	x8	x1		x1		x1		x1								
6h	x8	x4		x1							x1		x1		x1	
7h	x8	x4		x4												
8h	x8	x8														

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-29. 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	<b>Reserved</b>	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 r2.0</i> .	RO	Yes	0	
4	<b>Not valid</b>	Upstream	RsvdP	No	0
	<b>Link Disable</b> 1 = Places the Link on the corresponding PEX 8619 downstream Port to the <i>Disabled</i> Link Training state	Downstream	RW	Yes	0
5	<b>Not valid</b> Always read as 0.	Upstream	RsvdP	No	0
	<b>Retrain Link</b> For PEX 8619 Ports, always returns 0 when read; however, software is allowed to write this register. Writing 1 to this bit causes the corresponding PEX 8619 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> When Set, 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	

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-29. 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>Reserved</i> Read and Writable only when the <b>Link Capability</b> register <i>Clock Power Management Capable</i> bit is Set. The PEX 8619 does <i>not support</i> removal of the Reference Clock in the L1 and L2/L3 Ready Link PM states.		RsvdP	No	0
9	<b>Hardware-Autonomous Width Disable</b> <i>Reserved</i>		RsvdP	No	0
10	<i>Reserved</i> <b>Link Bandwidth Management Interrupt Enable</b> 0 = Disables interrupt generation 1 = Enables generation of an interrupt, to indicate that the <b>Link Status</b> register <i>Link Bandwidth Management Status</i> bit (offset 78h[30]) has been Set	Upstream	RsvdP	No	0
11	<i>Reserved</i> <b>Link Autonomous Bandwidth Interrupt Enable</b> 0 = Disables interrupt generation 1 = Enables generation of an interrupt, to indicate that the <b>Link Status</b> register <i>Link Autonomous Bandwidth Status</i> bit (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. 0001b = 2.5 GT/s Link speed 0010b = 5.0 GT/s Link speed All other encodings are <i>reserved</i> . The value in this field is undefined when the Link is not up.		RO	No	0001b
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. 00_0000b = Link is down (default) 00_0001b = x1 or Port is in the <i>DL_Down</i> state 00_0010b = x2 00_0100b = x4 00_1000b = x8 All other encodings are <i>not supported</i> .		RO	No	00_0000b
26	<i>Reserved</i>		RsvdP	No	0
27	<i>Reserved</i> <b>Link Training</b> 1 = Indicates that the corresponding PEX 8619 downstream Port requested Link training, and the Link training is in-progress or about to start	Upstream	RsvdP	No	0
Downstream					
			RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-29. 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 8619 uses an independent clock 1 = Indicates that the PEX 8619 uses the same physical Reference Clock that the platform provides on the connector		HwInit	Yes	0
29	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Data Link Layer Link Active</b> When Set, and the <b>Link Capability</b> register <i>Data Link Layer Link Active Reporting Capable</i> bit (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	<b>Reserved</b>	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 <b>Link Control</b> register <i>Retrain Link</i> bit (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	<b>Reserved</b>	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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-30. 7Ch Slot Capability  
(Downstream Ports; Upstream Port Always Reads 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> 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 and/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 8619 has detected that an external I <sup>2</sup> C I/O Expander is present.				
0	<b>Reserved</b> <b>Attention Button Present</b> 0 = Attention Button is not implemented 1 = Attention Button is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Upstream	RsvdP	No	0
0	<b>Reserved</b> <b>Power Controller Present</b> 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 slot of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander. The Controller powers up the slot when the Manually operated Retention Latch (MRL) is closed and the <b>Slot Control</b> register <i>Power Controller Control</i> bit (offset 80h[10]) is Cleared. Otherwise, if bit 2 ( <i>MRL Sensor Present</i> ) is disabled (Cleared), the MRL's position has no effect on powering up the slot.	Downstream	RO	Yes	1
1	<b>Reserved</b> <b>MRL Sensor Present</b> 0 = MRL Sensor is not implemented. MRL position is “Don’t Care” for that slot. 1 = MRL Sensor is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander. The PEX 8619 senses whether the MRL is open or closed for a slot. MRL should be Low for power-on for that slot.	Upstream	RsvdP	No	0
1	<b>Reserved</b> <b>Attention Indicator Present</b> 0 = Attention Indicator is not implemented 1 = Attention Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Downstream	RO	Yes	1
2	<b>Reserved</b> <b>MRL Sensor Present</b> 0 = MRL Sensor is not implemented. MRL position is “Don’t Care” for that slot. 1 = MRL Sensor is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander. The PEX 8619 senses whether the MRL is open or closed for a slot. MRL should be Low for power-on for that slot.	Upstream	RsvdP	No	0
2	<b>Reserved</b> <b>Attention Indicator Present</b> 0 = Attention Indicator is not implemented 1 = Attention Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Downstream	RO	Yes	1
3	<b>Reserved</b> <b>Attention Indicator Present</b> 0 = Attention Indicator is not implemented 1 = Attention Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Upstream	RsvdP	No	0
3	<b>Reserved</b> <b>Attention Indicator Present</b> 0 = Attention Indicator is not implemented 1 = Attention Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Downstream	RO	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-30. 7Ch Slot Capability  
(Downstream Ports; Upstream Port Always Reads 0) (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> 0 = Power Indicator is not implemented 1 = Power Indicator is implemented on the slot chassis of the corresponding PEX 8619 Transparent downstream Port with an I <sup>2</sup> C I/O Expander	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Hot Plug Surprise</b> 0 = No device in the corresponding PEX 8619 downstream Port (with an I <sup>2</sup> C I/O Expander) slot is removed from the system without prior notification 1 = Device in the corresponding PEX 8619 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> 0 = Corresponding PEX 8619 downstream Port slot is not capable of supporting Hot Plug operations 1 = Corresponding PEX 8619 Transparent downstream Port (with an I <sup>2</sup> C I/O Expander) 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 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-30. 7Ch Slot Capability  
(Downstream Ports; Upstream Port Always Reads 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
	<b>Reserved</b>	Upstream	RsvdP	No	00b
16:15	<b>Slot Power Limit Scale</b> The maximum power supplied by the corresponding PEX 8619 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>  <b>Electromechanical Interlock Present</b> This bit is valid for Serial Hot Plug Ports that have an I/O Expander. 0 = Electromechanical Interlock is not implemented on the chassis for this slot 1 = Electromechanical Interlock is implemented on the chassis for this slot	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
18	<b>No Command Completed Support</b> <b>Reserved</b>		RsvdP	No	0
31:19	<b>Reserved</b>  <b>Physical Slot Number</b> 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.	Upstream	RsvdP	No	0-0h
	<b>Bit(s)</b> <b>Description/Function</b>				
22:19	Port Numbers 0 through 15				
26:23	Loaded from I <sup>2</sup> C I/O Expander				
31:27	<b>Reserved</b>				

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-31. 80h Slot Status and Control****(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Slot Control</b>					
<p><b>Note:</b> To change the values of the <a href="#">MRL Sensor Changed Enable</a> (bit 2), <a href="#">Power Fault Detector Enable</a> (bit 1), and/or <a href="#">Attention Button Pressed Enable</a> (bit 0) bits, the corresponding bit(s) in the <a href="#">Slot Capability</a> register (offset 7Ch[2:0], respectively) must be Set first.</p>					
0	<b>Reserved</b> <b>Attention Button Pressed Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 Device PM state ( <a href="#">PCI Power Management Status and Control</a> register <a href="#">Power State</a> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot Device PM state (offset 44h[1:0]. are both Set), for an <a href="#">Attention Button Pressed</a> event on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	Upstream Offset 7Ch[0]=0	Downstream RO	No	0
0		Offset 7Ch[0]=1	Downstream RW	Yes	0
1	<b>Reserved</b> <b>Power Fault Detector Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 Device PM state ( <a href="#">PCI Power Management Status and Control</a> register <a href="#">Power State</a> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot Device PM state (offset 44h[1:0], are both Set), for a <a href="#">Power Fault Detected</a> event on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	Upstream Offset 7Ch[1]=0	Downstream RO	No	0
1		Offset 7Ch[1]=1	Downstream RW	Yes	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-31. 80h Slot Status and Control  
(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (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 Changed Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 Device PM state ( <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 Device PM state (offset 44h[1:0], are both Set), for an <b>MRL Sensor Changed</b> event on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	Offset 7Ch[2]=0	Downstream	RO	No 0
3	<b>Not valid</b>	Upstream	RsvdP	No	0
	<b>Presence Detect Changed Enable</b> A Presence Detect Changed event is triggered by either the SerDes Receiver Detect ( <b>Even/Odd Port Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 200h[31:24], and Odd Ports, offset 204h[31:16])) or Hot Plug PRSNT# input (from external I <sup>2</sup> C I/O Expander) on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port. 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 Device PM state ( <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 Device PM state (offset 44h[1:0], are both Set), for a <b>Presence Detect Changed</b> event on the corresponding PEX 8619 downstream Port	Offset 7Ch[2]=1	Downstream	RW	Yes 0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-31. 80h Slot Status and Control****(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (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 Serial Hot Plug Controller on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	Downstream	RW	Yes	0
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Hot Plug Interrupt Enable</b> 0 = Function is disabled 1 = Enables an interrupt on enabled Hot Plug/Link State events for the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port	Downstream	RW	Yes	0
7:6	<b>Reserved</b>	Upstream	RsvdP	No	00b
	<b>Attention Indicator Control</b> Controls the Attention Indicator on the corresponding PEX 8619 downstream Port slot. Reads return the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-31. 80h Slot Status and Control  
(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
	<b>Reserved</b>	Upstream	RsvdP	No	00b
9:8	<p><b>Power Indicator Control</b> Controls the Power Indicator on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot. Reads return the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port Power Indicator's current state.</p> <p>Writing a non-zero value triggers a <a href="#">Command Completed</a> 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.</p> <p>00b = <b>Reserved</b> – Writes are ignored 01b = Turns On indicator to constant On state 10b = Causes indicator to blink 11b = Turns Off indicator</p>	Downstream	RW	Yes	11b (MRL open) 01b (MRL closed)
10	<p><b>Reserved</b></p> <p><b>Power Controller Control</b> Controls the Power Controller on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot.</p> <p>0 = Turns On the Power Controller; requires some delay to be effective 1 = Turns Off the Power Controller</p>	Upstream	RsvdP	No	0
11	<p><b>Reserved</b></p> <p><b>Electromechanical Interlock Control</b> This bit is valid for Serial Hot Plug Ports that have an I/O Expander. 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 0.</p>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
12	<p><b>Not valid</b></p> <p><b>Data Link Layer State Changed Enable</b> Enables software notification with an interrupt if the Port is in the D0 Device PM state (<a href="#">PCI Power Management Status and Control</a> register <a href="#">Power State</a> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot Device PM state (offset 44h[1:0], are both Set), when the <a href="#">Link Status</a> register <a href="#">Data Link Layer Link Active</a> bit (offset 78h[29]) is changed.</p>	Downstream Serial Hot Plug-enabled	RW	Yes	0
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-31. 80h Slot Status and Control****(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (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 = Attention Button of the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot was pressed	Downstream	RW1C	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Fault Detected</b> 1 = Power Controller of the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot detected a Power Fault at the slot	Downstream	RW1C	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>MRL Sensor Changed</b> 1 = MRL Sensor state change was detected on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot	Downstream	RW1C	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Presence Detect Changed</b> A <b>Presence Detect Changed</b> event is triggered by either the SerDes Receiver Detect ( <b>Even/Odd Port Receiver Detect Status</b> register <i>Receiver Detected on Lane x bits</i> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>200h[31:24]</b> , and Odd Ports, offset <b>204h[31:16]</b> )) or Hot Plug PRSNT# input (from external I <sup>2</sup> C I/O Expander) on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port. Write 1 to Clear. 1 = Value reported in bit <b>22 (Presence Detect State)</b> changed	Downstream	RW1C	Yes	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-31. 80h Slot Status and Control  
(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (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 = Serial Hot Plug Controller on the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port slot 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>Reserved</b>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
	<b>MRL Sensor State</b> Reveals the corresponding PEX 8619 Serial Hot Plug-capable Transparent downstream Port MRL Sensor's current state. 0 = MRL Sensor is closed 1 = MRL Sensor is open	Downstream Serial Hot Plug-enabled	RO	No	0
22	<b>Not valid</b>	Upstream	RsvdP	No	0
	<b>Presence Detect State</b> For downstream Ports that implement slots, indicates the presence of an adapter in the slot, reflected by the logical OR of the corresponding downstream Port's SerDes Receiver Detect ( <b>Even/Odd Port Receiver Detect</b> )	Downstream, Offset 68h[24]=1	RO	No	0
	<b>Status</b> register <i>Receiver Detected on Lane x</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 200h[31:24], and Odd Ports, offset 204h[31:16])), and, if present, the PRSNT# input on the external I/O Expander for the Serial Hot Plug-enabled Port. Hardwired to 1 when the <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				
	<b>Reserved</b>	Downstream, Offset 68h[24]=0	RO	No	1
	<b>Electromechanical Interlock Status</b> This bit is valid for Serial Hot Plug Ports that have an I/O Expander. When an Electromechanical Interlock is implemented, indicates the Electromechanical Interlock's current status. 0 = Electromechanical Interlock is disengaged 1 = Electromechanical Interlock is engaged				
23	<b>Reserved</b>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
	<b>Electromechanical Interlock Status</b> This bit is valid for Serial Hot Plug Ports that have an I/O Expander. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-31. 80h Slot Status and Control****(Transparent Downstream Serial Hot Plug-Enabled Ports; Upstream Port Always Reads 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<i>Not valid</i>	Upstream	RsvdP	No	0
	<b>Data Link Layer State Changed</b> In response to a <b>Data Link Layer State Changed</b> event, software must read the <b>Link Status</b> register <i>Data Link Layer Link Active</i> bit (offset 78h[29]), to determine whether the Link is active before initiating Configuration Requests to the device.  1 = Value reported in the <b>Link Status</b> register <i>Data Link Layer Link Active</i> bit changed	Downstream	RW1C	Yes	0
31:25	<b>Reserved</b>		RsvdZ	No	0-0h

**Register 14-32. 8Ch Device Capability 2****(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:0	<b>Reserved</b>		RsvdP	No	0-0h
5	<b>Reserved</b>	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
31:6	<b>Reserved</b>		RsvdP	No	0-0h

**Register 14-33. 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	<b>Reserved</b>		RsvdP	No	0-0h
5	<b>Reserved</b>	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
15:6	<b>Reserved</b>		RsvdP	No	0-0h
<b>Device Status 2</b>					
31:16	<b>Reserved</b>		RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-34. 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> 0001b = 2.5 GT/s Link speed is supported 0010b = 5.0 GT/s Link speed is supported All other encodings are <i>reserved</i> .	RWS	Yes	0010b	
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	<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.	Upstream	RWS	Yes	000b
	<i>Reserved</i>	Downstream	RsvdP	Yes	0
10	<b>Enter Modified Compliance</b> Intended for debug and compliance testing only. Valid only for Function 0 of the upstream Port.	Upstream	RWS	Yes	0
	<i>Reserved</i>	Downstream	RsvdP	Yes	0
11	<b>Compliance SOS</b> 1 = LTSSM must periodically send SKIP Ordered-Sets between sequences when sending the Compliance Pattern or Modified Compliance Pattern	Upstream	RWS	Yes	0
	<i>Reserved</i>	Downstream	RsvdP	Yes	0
12	<b>Compliance De-Emphasis</b> Sets the de-emphasis level in the <i>Polling.Compliance</i> state, if the entry occurred due to bit 4 ( <i>Enter Compliance</i> ) being Set.	Upstream	RWS	Yes	0
	<i>Reserved</i>	Downstream	RsvdP	Yes	0
15:13	<i>Reserved</i>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-34. 98h Link Status and Control 2  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	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. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at 2.5 GT/s)	RO	Yes	0 (5.0 GT/s) 1 (2.5 GT/s)	
31:17	<b>Reserved</b>	RsvdP	No	0-0h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 14.10 Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)

This section details the Subsystem ID and Subsystem Vendor ID Capability registers. Table 14-11 defines the register map.

**Table 14-11. 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 14-35. 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> Detects the 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 8619 Port's Capabilities list The PEX 8619 Extended Capabilities list starts at offset 100h.	RO	Yes	00h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 14-36. 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 <b>Vendor ID</b> (offset 00h[15:0]) identifies the manufacturer of the PEX 8619, 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 8619 Ports.	RO	Yes	10B5h
31:16	<b>Subsystem ID</b> The <b>Device ID</b> (offset 00h[31:16]) identifies the PEX 8619, 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 8619 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	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.11 Device Serial Number Extended Capability Registers (Offsets 100h – 134h)

This section details the Device Serial Number Extended Capability registers. [Table 14-12](#) defines the register map.

**Table 14-12. Device Serial Number 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 (FB4h)	Capability Version (1h)	PCI Express Extended Capability ID (0003h)	100h
		Serial Number (Lower DW)	104h
		Serial Number (Upper DW)	108h
		<i>Reserved</i>	10Ch – 134h

**Register 14-37. 100h Device Serial Number 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 0003h, as required by the <i>PCI Express Base r2.0</i> .	RO	Yes	0003h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</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	<b>FB4h</b>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-38. 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 r2.0</i> , all switch PEX 8619 Ports must contain the same value; therefore, one physical register is shared by all PEX 8619 Ports. (Refer to <a href="#">Table 14-3</a> .) The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64 <sup>TM</sup> ). The lower 24 bits are the Company ID value assigned by the IEEE registration authority, and the upper 40 bits are the Extension ID assigned by the identified Company.	RO	Yes	B5DF_0E00h

**Register 14-39. 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 r2.0</i> , all switch Ports must contain the same value; therefore, one physical register is shared by all PEX 8619 Ports. (Refer to <a href="#">Table 14-3</a> .) The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64 <sup>TM</sup> ). The lower 24 bits are the Company ID value assigned by the IEEE registration authority, and the upper 40 bits are the Extension ID assigned by the identified Company.	RO	Yes	BA_8601_10h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.12 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 upstream Port/NT Port Link Interface **Power Budget Data** register (offset 140h), write 5 into the upstream Port/NT Port Link Interface **Data Select** register **Data Select** field (offset 13Ch[7:0]), then write the value into the upstream Port/NT Port Link Interface **Power Budget Data** register. **Table 14-13** defines the register map.

**Table 14-13. Power Budget Extended Capability Register Map  
(Upstream Port, and also the NT Port Link 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				
Next Capability Offset (148h)	Capability Version (1h)	PCI Express Extended Capability ID (0004h)			138h
	<i>Reserved</i>		Data Select		13Ch
		Power Budget Data			140h
		Power Budget Capability			144h

**Register 14-40. 138h Power Budget Extended Capability Header  
(Upstream Port, and also the NT Port Link Interface)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	PCI Express Extended Capability ID Program to 0004h, as required by the <i>PCI Express Base r2.0</i> .	Upstream	RO	Yes	0004h
	<i>Reserved</i>	Downstream	RsvdP	No	0000h
19:16	Capability Version Program to 1h, as required by the <i>PCI Express Base r2.0</i> .	Upstream	RO	Yes	1h
	<i>Reserved</i>	Downstream	RsvdP	No	0h
31:20	Next Capability Offset Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	Upstream	RO	Yes	148h
	<i>Reserved</i>	Downstream	RsvdP	No	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-41. 13Ch Data Select  
(Upstream Port, and also the NT Port Link Interface)**

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 upstream Port/NT Port Link Interface <b>Power Budget Data</b> registers, per 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 14-42. 140h Power Budget Data  
(Upstream Port, and also the NT Port Link Interface)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> Eight registers per upstream Port/NT Port Link Interface 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 (Upstream Port, and also the NT Port Link Interface, offset 13Ch[7:0]).					
7:0	<b>Base Power</b> Eight registers per upstream Port/NT Port Link Interface. 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
	<b>PM Sub-State</b> 000b = Power Management sub-state of the operating condition being described	Upstream	RO	Yes	000b
12:10	<b>Reserved</b>	Downstream	RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-42. 140h Power Budget Data  
(Upstream Port, and also the NT Port Link Interface) (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 Device PM state 11b = D3 Device PM state All other encodings are <i>reserved</i> .	Upstream	RO	Yes	00b
	<b>Reserved</b>	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
	<b>Reserved</b>	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
	<b>Reserved</b>	Downstream	RsvdP	No	000b
31:21	<b>Reserved</b>		RsvdP	No	0-0h

**Register 14-43. 144h Power Budget Capability  
(Upstream Port, and also the NT Port Link Interface)**

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
	<b>Reserved</b>	Downstream	RsvdP	No	0
31:1	<b>Reserved</b>			RsvdP	No
					0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.13 Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)

This section details the Virtual Channel Extended Capability registers, which are duplicated for each Port. [Table 14-14](#) defines the register map for one Port.

**Table 14-14. Virtual Channel 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 (448h, 950h, or 520h)	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> )			154h
VC0 Resource Capability			158h
VC0 Resource Control			15Ch
VC0 Resource Status			160h
VC1 Resource Capability			164h
VC1 Resource Control			168h
VC1 Resource Status			16Ch
<i>Reserved</i>			170h –
WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh)			1A4h
			1A8h
			...
			1BCh

**Register 14-44. 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 r2.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</i> .	RO	No	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>Vendor-Specific Extended Capability</b> structure, offset 448h.	RO	No	448h
	Next extended capability is the <b>Vendor-Specific Extended Capability 2</b> structure, offset 950h.	RO	No	950h
	Next extended capability is the <b>ACS Extended Capability</b> structure, offset 520h.	RO	No	520h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-45. 14Ch Port VC Capability 1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Extended VC Counter</b> 000b = PEX 8619 Port supports only the default Virtual Channel (VC0) 001b = PEX 8619 Port supports <b>one</b> extended Virtual Channel (VC1) All other encodings are <b>reserved</b> .	RO	Yes	001b
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 the default, VC0) that belong to the Low-Priority VC group for this PEX 8619 Port. Both VC0 and VC1 are assigned to the Low-Priority Virtual Channel group, by default. 000b = For this PEX 8619 Port, only the default VC, VC0, belongs to the Low-Priority Virtual Channel group. VC1 has higher priority than VC0. 001b = For this PEX 8619 Port, both VC0 and VC1 belong to the Low-Priority Virtual Channel group. All other encodings are <b>reserved</b> .	RO	Yes	001b
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> 10b = Port Arbitration Table entry size is 4 bits	RO	Yes	10b
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-46. 150h Port VC Capability 2  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>VC Arbitration Capability</b> Indicates the type of VC arbitration supported by the device for the LPVC (Low-Priority Extended VC) group. 0 = Indicates that the Round-Robin (Hardware-Fixed) Arbitration scheme is not supported 1 = Indicates that the Round-Robin (Hardware-Fixed) Arbitration scheme is supported	RO	Yes	1
31:1	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-47. 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> Writing 1 updates the Port's VC Arbitration Table. Reads always return 0.	RsvdP	No	0
<b>Port VC Status</b>				
16	<b>VC Arbitration Table Status</b> <i>Reserved</i>	RsvdP	No	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-48. 158h VC0 Resource Capability  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Port Arbitration Capability</b> When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0.  Bit 0 = 1 – Non-configurable Round-Robin (Hardware-Fixed) Arbitration Bit 1 = 1 – Weighted Round-Robin (WRR) arbitration with 32 Phases	RO	No	10b
		RO	No	01b
14:2	<b>Reserved</b>	RsvdP	No	0-0h
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>Reserved</i>	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 14.13.1</a> for further details.)	RO	No	06h
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0.  00h = Non-configurable Round-Robin (Hardware-Fixed) Arbitration 06h = Weighted Round-Robin (WRR) arbitration with 32 Phases	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-49. 15Ch VC0 Resource Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 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 0.	RW1S	Yes	0
19:17	<b>Port Arbitration Select</b> Selects the Port Arbitration type for the corresponding PEX 8619 Port. 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.  When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0.  0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	001b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID</b> Defines the Port's VC0 ID code. This bit is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable</b> 0 = Not allowed 1 = Enables the Port's default VC, VC0	RO	No	1

**Register 14-50. 160h VC0 Resource Status  
(All Ports)**

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> 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]) 1 = Port Arbitration Table entry was written to by software	RO	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	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-51. 164h VC1 Resource Capability  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Port Arbitration Capability</b> VC1 Requests use the Port arbitration method defined by the <b>VC0 Resource Capability</b> register <i>Port Arbitration Capability</i> field (offset 158h[1:0]) for VC0. 01h = Non-configurable Hardware-Fixed Port arbitration – the only value supported	RO	No	01h
14:8	<b>Reserved</b>	RsvdP	No	0-0h
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>Reserved</i>	RsvdP	No	000_0000b
23	<b>Reserved</b>	RsvdP	No	0
31:24	<b>Port Arbitration Table Offset</b> <i>Not used</i> VC1 Requests use the Port arbitration method defined by the <b>VC0 Resource Capability</b> register <i>Port Arbitration Capability</i> field (offset 158h[1:0]) for VC0.	RsvdP	No	00h

**Register 14-52. 168h VC1 Resource Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1.	RO	No	00h
7:1	Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.		Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table</b> <i>Not supported</i> Hardware writable and software readable.	RsvdP	No	0
19:17	<b>Port Arbitration Select</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID</b> Defines the Port's VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable</b> 0 = Disables the Port's VC1 1 = Enables the Port's VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-53. 16Ch VC1 Resource Status  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b><i>Reserved</i></b>	RsvdP	No	0000h
16	<b>Port Arbitration Table Status</b> <i>Not supported</i>	RO	No	0
17	<b>VC1 Negotiation Pending</b> 0 = Port's VC1 negotiation is complete 1 = Port's VC1 initialization or disabling is pending	RO	Yes	1
31:18	<b><i>Reserved</i></b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.13.1 WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh)

This section details the WRR Port Arbitration Table registers. Port Arbitration Table phases are used to determine Port weighting during “Weighted Round-Robin with 32 Phases” Port arbitration.

**Table 14-15** defines the register map. The numbers along the top of the table indicate the 4-bit fields of each 32-bit register. There are 32 phases, and any active Port Number can go into each Port  $x$ , Phase  $x$  box.

**Note:** *The Port Arbitration Table is used only when Weighted Round-Robin (WRR) with 32-phase Port Arbitration is selected. The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port. If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and reserved on all other Ports.*

**Table 14-15. WRR Port Arbitration Table Register Map  
(When WRR is enabled)**

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 $x$ , Phase 7	Port $x$ , Phase 6	Port $x$ , Phase 5	Port $x$ , Phase 4	Port $x$ , Phase 3	Port $x$ , Phase 2	Port $x$ , Phase 1	Port $x$ , Phase 0	1A8h
Port $x$ , Phase 15	Port $x$ , Phase 14	Port $x$ , Phase 13	Port $x$ , Phase 12	Port $x$ , Phase 11	Port $x$ , Phase 10	Port $x$ , Phase 9	Port $x$ , Phase 8	1ACh
Port $x$ , Phase 23	Port $x$ , Phase 22	Port $x$ , Phase 21	Port $x$ , Phase 20	Port $x$ , Phase 19	Port $x$ , Phase 18	Port $x$ , Phase 17	Port $x$ , Phase 16	1B0h
Port $x$ , Phase 31	Port $x$ , Phase 30	Port $x$ , Phase 29	Port $x$ , Phase 28,	Port $x$ , Phase 27	Port $x$ , Phase 26	Port $x$ , Phase 25	Port $x$ , Phase 24	1B4h
<i>Reserved</i>								1B8h – 1BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-54. 1A8h WRR Port Arbitration Table Phases 0 to 7  
(When WRR is enabled)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port.				
If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and <i>reserved</i> on all other Ports.				
3:0	<b>Port Arbitration Table Phase 0</b>	RW	Yes	0h
7:4	<b>Port Arbitration Table Phase 1</b>	RW	Yes	1h
11:8	<b>Port Arbitration Table Phase 2</b>	RW	Yes	2h
15:12	<b>Port Arbitration Table Phase 3</b>	RW	Yes	3h
19:16	<b>Port Arbitration Table Phase 4</b>	RW	Yes	4h
23:20	<b>Port Arbitration Table Phase 5</b>	RW	Yes	5h
27:24	<b>Port Arbitration Table Phase 6</b>	RW	Yes	6h
31:28	<b>Port Arbitration Table Phase 7</b>	RW	Yes	7h

**Register 14-55. 1ACh WRR Port Arbitration Table Phases 8 to 15  
(When WRR is enabled)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port.				
If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and <i>reserved</i> on all other Ports.				
3:0	<b>Port Arbitration Table Phase 8</b>	RW	Yes	8h
7:4	<b>Port Arbitration Table Phase 9</b>	RW	Yes	9h
11:8	<b>Port Arbitration Table Phase 10</b>	RW	Yes	Ah
15:12	<b>Port Arbitration Table Phase 11</b>	RW	Yes	Bh
19:16	<b>Port Arbitration Table Phase 12</b>	RW	Yes	Ch
23:20	<b>Port Arbitration Table Phase 13</b>	RW	Yes	Dh
27:24	<b>Port Arbitration Table Phase 14</b>	RW	Yes	Eh
31:28	<b>Port Arbitration Table Phase 15</b>	RW	Yes	Fh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-56. 1B0h WRR Port Arbitration Table Phases 16 to 23  
(When WRR is enabled)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port.				
If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and <i>reserved</i> on all other Ports.				
3:0	<b>Port Arbitration Table Phase 16</b>	RW	Yes	0h
7:4	<b>Port Arbitration Table Phase 17</b>	RW	Yes	1h
11:8	<b>Port Arbitration Table Phase 18</b>	RW	Yes	2h
15:12	<b>Port Arbitration Table Phase 19</b>	RW	Yes	3h
19:16	<b>Port Arbitration Table Phase 20</b>	RW	Yes	4h
23:20	<b>Port Arbitration Table Phase 21</b>	RW	Yes	5h
27:24	<b>Port Arbitration Table Phase 22</b>	RW	Yes	6h
31:28	<b>Port Arbitration Table Phase 23</b>	RW	Yes	7h

**Register 14-57. 1B4h WRR Port Arbitration Table Phases 24 to 31  
(When WRR is enabled)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port.				
If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and <i>reserved</i> on all other Ports.				
3:0	<b>Port Arbitration Table Phase 24</b>	RW	Yes	8h
7:4	<b>Port Arbitration Table Phase 25</b>	RW	Yes	9h
11:8	<b>Port Arbitration Table Phase 26</b>	RW	Yes	Ah
15:12	<b>Port Arbitration Table Phase 27</b>	RW	Yes	Bh
19:16	<b>Port Arbitration Table Phase 28</b>	RW	Yes	Ch
23:20	<b>Port Arbitration Table Phase 29</b>	RW	Yes	Dh
27:24	<b>Port Arbitration Table Phase 30</b>	RW	Yes	Eh
31:28	<b>Port Arbitration Table Phase 31</b>	RW	Yes	Fh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 14.14 Device-Specific Registers (Offsets 1C0h – 51Ch)

This section details the Device-Specific registers located at offsets 1C0h through 51Ch. Device-Specific registers are unique to the PEX 8619 and not referenced in the *PCI Express Base r2.0*. Table 14-16 defines the register map.

Other Device-Specific registers are detailed in [Section 14.16, “Device-Specific Registers \(Offsets 530h – F8Ch\).”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 14-16. Device-Specific Register Map (Offsets 1C0h – 51Ch)**

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
		...
		1FCh
		200h
		...
		25Ch
		260h
		...
		26Ch
		270h –
		288h
		28Ch
		290h
		...
		2C4h
		2C8h
		...
		304h
		308h
		...
		340h
		344h
		348h
		...
		444h
Next Capability Offset (950h or 520h)	1h	PCI Express Extended Capability ID (000Bh)
		448h
		...
		51Ch
Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)		
Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)		
Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch)		
<b>Factory Test Only/Reserved</b>		
Device-Specific Registers – Miscellaneous Control (Offset 28Ch)		
Device-Specific Registers – I <sup>2</sup> C Slave Interface (Offsets 290h – 2C4h)		
Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h)		
Device-Specific Registers – I/O CAM (Offsets 308h – 340h)		
Device-Specific Registers – SMBus Slave Interface (Offset 344h)		
Device-Specific Registers – Address-Mapping CAM (Offsets 348h – 444h)		
Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability (Offsets 448h – 51Ch)		

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.14.1 Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)

This section details the Device-Specific Error Checking and Debug registers. [Table 14-17](#) defines the register map.

**Table 14-17. 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	
	Device-Specific Error Status for Egress ECC Error	1C0h
	Device-Specific Error Mask for Egress ECC Error	1C4h
	ECC Error Check Disable	1C8h
	Error Handler 32-Bit Error Status	1CCh
	Error Handler 32-Bit Error Mask	1D0h
	<i>Factory Test Only</i>	1D4h
<i>Reserved</i>	Clock Enable	1D8h
	Debug Control	1DCh
	Power Management Hot Plug User Configuration	1E0h
	Egress Control and Status	1E4h
	Bad TLP Counter	1E8h
	Bad DLLP Counter	1ECh
	<i>Reserved</i>	1F0h
<i>Reserved</i>	Software Lane Status	1F4h
	ACK Transmission Latency Limit	1F8h
	<i>Factory Test Only</i>	1FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-58. 1C0h Device-Specific Error Status for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
ECC is checked by the egress Port, and any ECC error is reported by the egress Half Station (containing the egress Port), when that Port reads the Packet data from the ingress Half-Station RAM. <i>For example</i> , when Port 4 receives a packet, the data is stored in Half-Station 1 RAM; then, if the egress Port is in Half-Station 0 and an ECC error is detected when the Port reads the Half-Station 1 RAM, the error is flagged in the Port 0 register bit that reflects Half-Station 1 error status.				
<i>Note:</i> The bits in this register can be masked by their respective <b>Device-Specific Error Mask for Egress ECC Error</b> register bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1C4h).				
0	<b>Payload Link List RAM Instance 0 1-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No overflow is detected 1 = 1-bit Soft error is detected	RW1CS	Yes	0
1	<b>Payload Link List RAM Instance 0 1-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No overflow is detected 1 = 1-bit Soft error is detected	RW1CS	Yes	0
2	<b>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No overflow is detected 1 = Read detected a 2-bit Soft error	RW1CS	Yes	0
3	<b>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No overflow is detected 1 = Read detected a 2-bit Soft error	RW1CS	Yes	0
4	<b>Ingress Link List RAM Read Detected 1-Bit ECC Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Read detected a 1-Bit ECC error	RW1CS	Yes	0
5	<b>Ingress Link List RAM Read Detected 2-Bit ECC Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Read detected a 2-Bit ECC error	RW1CS	Yes	0
6	<b>Ingress Link List RAM Read Detected 1-Bit ECC Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Read detected a 1-Bit ECC error	RW1CS	Yes	0
7	<b>Ingress Link List RAM Read Detected 2-Bit ECC Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Read detected a 2-Bit ECC error	RW1CS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-58. 1C0h Device-Specific Error Status for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
9	<b>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
10	<b>Factory Test Only</b>	RO	No	0
11	<b>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
12	<b>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
13	<b>Factory Test Only</b>	RO	No	0
14	<b>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
15	<b>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-58. 1C0h Device-Specific Error Status for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<i>Factory Test Only</i>	RO	No	0
17	<b>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
18	<b>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
19	<i>Factory Test Only</i>	RO	No	0
20	<b>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
21	<b>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
22	<i>Factory Test Only</i>	RO	No	0
23	<b>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-58. 1C0h Device-Specific Error Status for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = 1-Bit Soft Error Tracking Counter overflow is detected	RW1CS	Yes	0
25	<b>Factory Test Only</b>	RO	No	0
26	<b>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> <b>Reserved</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
27	<b>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> <b>Reserved</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
28	<b>Factory Test Only</b>	RO	No	0
29	<b>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> <b>Reserved</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
30	<b>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> <b>Reserved</b> 0 = No error is detected 1 = Read detected a 2-Bit Soft error	RW1CS	Yes	0
31	<b>Factory Test Only</b>	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-59. 1C4h Device-Specific Error Mask for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

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>Device-Specific Error Status for Egress ECC Error</b> register bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1C0h).				
0	<b>Payload Link List RAM Instance 0 1-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Payload Link List RAM Instance 0 1-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
1	<b>Payload Link List RAM Instance 0 1-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Payload Link List RAM Instance 0 1-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
2	<b>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
3	<b>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Payload Link List RAM Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
4	<b>Ingress Link List RAM Read Detected 1-Bit ECC Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM Read Detected 1-Bit ECC Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
5	<b>Ingress Link List RAM Read Detected 2-Bit ECC Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM Read Detected 2-Bit ECC Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
6	<b>Ingress Link List RAM Read Detected 1-Bit ECC Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM Read Detected 1-Bit ECC Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
7	<b>Ingress Link List RAM Read Detected 2-Bit ECC Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM Read Detected 2-Bit ECC Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-59. 1C4h Device-Specific Error Mask for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
9	<b>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
10	<b>Factory Test Only</b>	RWS	Yes	1
11	<b>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
12	<b>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
13	<b>Factory Test Only</b>	RWS	Yes	1
14	<b>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
15	<b>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-59. 1C4h Device-Specific Error Mask for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<i>Factory Test Only</i>	RWS	Yes	1
17	<b>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
18	<b>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 0 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
19	<i>Factory Test Only</i>	RWS	Yes	1
20	<b>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
21	<b>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 0 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
22	<i>Factory Test Only</i>	RWS	Yes	1
23	<b>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-59. 1C4h Device-Specific Error Mask for Egress ECC Error  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 1 1-Bit Soft Error Tracking Counter Overflow Detected Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
25	<b>Factory Test Only</b>	RWS	Yes	1
26	<b>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
27	<b>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 0 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
28	<b>Factory Test Only</b>	RWS	Yes	1
29	<b>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1
30	<b>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Packet RAM 1 Instance 1 Read Detected 2-Bit Soft Error Status for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1
31	<b>Factory Test Only</b>	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-60. 1C8h ECC Error Check Disable  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
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	0	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	0	RWS	Yes	0
2	<b>Software Force Error Enable</b> 1 = <b>Correctable Error Status</b> and <b>Uncorrectable Error Status</b> registers (offsets <b>FC4h</b> and <b>FB8h</b> , respectively) change from RW1CS to RW		RWS	Yes	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-60. 1C8h ECC Error Check Disable  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Enable PEX_INTA# Interrupt Output(s) for Hot Plug or Link State Event-Triggered Interrupts</b> 0 = Hot Plug or Link State Event Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = Hot Plug or Link State Event Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)		RWS	Yes	0
5	<b>Enable PEX_INTA# Interrupt Output(s) for Device-Specific Error-Triggered Interrupts</b> 0 = Device-Specific Error Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = Device-Specific Error Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)		RWS	Yes	0
6	<b>Enable PEX_INTA# Interrupt Output(s) for GPIO-Generated Interrupts</b> 0 = General-Purpose Input/Output (GPIO) Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = GPIO Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)		RWS	Yes	0
7	<b>Enable PEX_INTA# Interrupt Output(s) for NT-Virtual Doorbell-Generated Interrupts</b> <i>This bit is valid only in NT mode.</i> Enables either PEX_INTA# or INTx Messages for NT-Virtual Doorbell interrupts (NT Port Virtual Interface, offsets C4Ch through C58h). 0 = NT-Virtual Doorbell Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = NT-Virtual Doorbell Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)	NT Port Virtual Interface, or Port 0 if Port 0 is the NT Port	RWS	Yes	0
8	<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:</i> This bit must remain Cleared, for compliance to specifications governing the MSI Capability.		RWS	Yes	0
31:9	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-61. 1CCh Error Handler 32-Bit Error Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> All errors in this register generate MSI/INTx interrupts, if enabled.					
The bits in this register can be masked by their respective <b>Error Handler 32-Bit Error Mask</b> register bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1D0h</b> , with the exception of bit 0, which is in all Ports).					
0	<b>Completion FIFO Overflow Status</b> 0 = No overflow is detected 1 = Completion FIFO Overflow is detected when a 4-deep Completion FIFO for ingress, or a 1-deep Completion FIFO for egress, overflows	All	RW1CS	Yes	0
1	<b>Reserved</b>	RsvdP	No	0	
2	<b>Factory Test Only</b>	RW1CS	No	0	
3	<b>Reserved</b>	RsvdP	No	0	
4	<b>Destination Queue Link List RAM 2-Bit Error</b> 0 = No error is detected 1 = Destination Queue Link List RAM 2-bit error is detected	RW1CS	Yes	0	
5	<b>Reserved</b>	RsvdP	No	0	
6	<b>Destination Queue Link List RAM 1-Bit Error Counter Overflow</b> 0 = No error is detected 1 = Destination Queue Link List RAM 1-bit error is detected	RW1CS	Yes	0	
7	<b>Reserved</b>	RsvdP	No	0	
8	<b>Source Queue Link List RAM 1-Bit Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Source Queue Link List RAM 1-bit error is detected	RW1CS	Yes	0	
9	<b>Source Queue Link List RAM 1-Bit Error Counter Overflow for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Source Queue Link List RAM 1-bit error is detected	RW1CS	Yes	0	
10	<b>Source Queue Link List RAM 2-Bit Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Source Queue Link List RAM 2-bit error is detected	RW1CS	Yes	0	
11	<b>Source Queue Link List RAM 2-Bit Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Source Queue Link List RAM 2-bit error is detected	RW1CS	Yes	0	
12	<b>Retry Buffer 1-Bit Error Counter Overflow</b> 0 = No error is detected 1 = Retry Buffer 1-bit error is detected	RW1CS	Yes	0	
13	<b>Retry Buffer 2-Bit ECC Error</b> 0 = No error is detected 1 = Retry Buffer 2-bit error is detected	RW1CS	Yes	0	
15:14	<b>Reserved</b>	RsvdP	No	00b	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-61. 1CCh Error Handler 32-Bit Error Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Descriptor RAM 1-Bit ECC Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Descriptor RAM 1-bit Error Counter overflow is detected	RW1CS	Yes	0	
17	<b>Descriptor RAM 1-Bit ECC Error Counter Overflow for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Descriptor RAM 1-bit error Counter overflow is detected	RW1CS	Yes	0	
18	<b>Descriptor RAM 2-Bit ECC Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = Descriptor RAM 2-bit error is detected	RW1CS	Yes	0	
19	<b>Descriptor RAM 2-Bit ECC Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = Descriptor RAM 2-bit error is detected	RW1CS	Yes	0	
20	<b>TLP ID RAM 2-Bit ECC Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = TLP ID RAM 2-bit ECC error is detected	RW1CS	Yes	0	
21	<b>TLP ID RAM 2-Bit ECC Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = TLP ID RAM 2-bit ECC error is detected	RW1CS	Yes	0	
23:22	<b>Reserved</b>	RsvdP	No	00b	
24	<b>TLP ID RAM 1-Bit ECC Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No error is detected 1 = TLP ID RAM 1-bit ECC Error Counter overflow is detected	RW1CS	Yes	0	
25	<b>TLP ID RAM 1-Bit ECC Error Counter Overflow for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No error is detected 1 = TLP ID RAM 1-bit ECC Error Counter overflow is detected	RW1CS	Yes	0	
31:26	<b>Reserved</b>	RsvdP	No	0-0h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-62. 1D0h Error Handler 32-Bit Error Mask  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> Error logging is enabled in this register, by default.					
The bits in this register can be used to mask their respective <b>Error Handler 32-Bit Error Status</b> register bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>1CCh</b> , with the exception of bit 0, which is in all Ports).					
0	<b>Completion FIFO Overflow Mask</b> 0 = If enabled, error generates MSI/INTx interrupt 1 = <i>Completion FIFO Overflow Status</i> bit is masked/disabled	All	RWS	Yes	1
1	<b>Reserved</b>	RsvdP	No	0	
2	<b>Factory Test Only</b>	RWS	Yes	1	
3	<b>Reserved</b>	RsvdP	No	0	
4	<b>Destination Queue Link List RAM 2-Bit Error Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Link List RAM 2-Bit Error</i> bit is masked/disabled	RWS	Yes	1	
5	<b>Reserved</b>	RsvdP	No	0	
6	<b>Destination Queue Link List RAM 1-Bit Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Link List RAM 1-Bit Error Counter Overflow</i> bit is masked/disabled	RWS	Yes	1	
7	<b>Reserved</b>	RsvdP	No	0	
8	<b>Source Queue Link List RAM 1-Bit Error Counter Overflow Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 1-Bit Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1	
9	<b>Source Queue Link List RAM 1-Bit Error Counter Overflow Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 1-Bit Error Counter Overflow for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1	
10	<b>Source Queue Link List RAM 2-Bit Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 2-Bit Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	1	
11	<b>Source Queue Link List RAM 2-Bit Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 2-Bit Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	1	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-62. 1D0h Error Handler 32-Bit Error Mask  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Retry Buffer 1-Bit Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Retry Buffer 1-Bit Error Counter Overflow</i> bit is masked/disabled	RWS	Yes	1	
13	<b>Retry Buffer 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Retry Buffer 2-Bit ECC Error</i> bit is masked/disabled	RWS	Yes	1	
15:14	<b>Reserved</b>	RsvdP	No	00b	
16	<b>Descriptor RAM 1-Bit ECC Error Counter Overflow Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Descriptor RAM 1-Bit ECC Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	0	
17	<b>Descriptor RAM 1-Bit ECC Error Counter Overflow Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Descriptor RAM 1-Bit ECC Error Counter Overflow for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	0	
18	<b>Descriptor RAM 2-Bit ECC Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>Descriptor RAM 2-Bit ECC Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes	0	
19	<b>Descriptor RAM 2-Bit ECC Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>Descriptor RAM 2-Bit ECC Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes	0	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-62. 1D0h Error Handler 32-Bit Error Mask****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>TLP ID RAM 2-Bit ECC Error Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>TLP ID RAM 2-Bit ECC Error for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes		1
21	<b>TLP ID RAM 2-Bit ECC Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>TLP ID RAM 2-Bit ECC Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes		1
23:22	<b>Reserved</b>	RsvdP	No		00b
24	<b>TLP ID RAM 1-Bit ECC Error Counter Overflow Mask for Ports 0, 2, 4, 6, 8, 10, 12, and 14</b> 0 = No effect on reporting activity 1 = <i>TLP ID RAM 1-Bit ECC Error Counter Overflow for Ports 0, 2, 4, 6, 8, 10, 12, and 14</i> bit is masked/disabled	RWS	Yes		1
25	<b>TLP ID RAM 2-Bit ECC Error Mask for Ports 1, 3, 5, 7, 9, 11, 13, and 15</b> 0 = No effect on reporting activity 1 = <i>TLP ID RAM 2-Bit ECC Error for Ports 1, 3, 5, 7, 9, 11, 13, and 15</i> bit is masked/disabled	RWS	Yes		1
31:26	<b>Reserved</b>	RsvdP	No		0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-63. 1D8h Clock Enable**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Ports are automatically enabled, according to the Port configuration defined by the STRAP_PORTCFG[3:0] inputs, which can be overridden by programming the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> ).				
An enabled Port can be selectively disabled, however, by Clearing the Port's <i>Port x Clock Enable</i> bit in this register. Port 0 must always remain enabled.				
<i>Note:</i> It is not possible to enable more Ports than the maximum specified for the device.				
0	<b>Port 0 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	1
1	<b>Port 1 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
2	<b>Port 2 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
3	<b>Port 3 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-63. 1D8h Clock Enable****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 4 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
5	<b>Port 5 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
6	<b>Port 6 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
7	<b>Port 7 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
8	<b>Port 8 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
9	<b>Port 9 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
10	<b>Port 10 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
11	<b>Port 11 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-63. 1D8h Clock Enable**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
13	<b>Port 13 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
14	<b>Port 14 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
15	<b>Port 15 Clock Enable</b> 0 = Disables 1 = Enables	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-64. 1DCh Debug Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default				
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)			
<i>Notes:</i> The first Configuration register programmed by the serial EEPROM must be the <b>Debug Control</b> register (Port 0, offset <b>1DCh</b> ), serial EEPROM locations 4h through 9h, as listed in <a href="#">Table 6-1, “Serial EEPROM Data.”</a>								
If Port 0 is a Legacy NT Port, this register is loaded from the NT Port Virtual Interface register offset 1DCh location in the serial EEPROM.								
3:0	<b>Factory Test Only</b>	RO	No	Fh				
4	<b>UPCFG Timer Enable</b> Reflects the <a href="#">STRAP_UPCFG_TIMER_EN#</a> input state. 0 = Upconfigure Timer is disabled 1 = Upconfigure Timer is enabled	RWS	Yes	0 (STRAP_UPCFG_TIMER_EN#=H) 1 (STRAP_UPCFG_TIMER_EN#=L)				
5	<b>SMBus Enable</b> Reflects the <a href="#">STRAP_SMBUS_EN#</a> input state. 0 = SMBus Slave interface is disabled for device configuration (I <sup>2</sup> C mode is enabled) 1 = SMBus Slave interface is enabled for device configuration (SMBus mode is enabled)	RWS	Yes	0 (STRAP_SMBUS_EN#=H) 1 (STRAP_SMBUS_EN#=L)				
6	<b>NT P2P Enable</b> Reflects the <a href="#">STRAP_NT_P2P_EN#</a> input state. This bit and its corresponding input must <b>not</b> be toggled at runtime. 0 = NT PCI-to-PCI bridge mode is disabled (STRAP_NT_P2P_EN#=H) (Legacy NT mode is enabled) 1 = NT PCI-to-PCI bridge mode is enabled (STRAP_NT_P2P_EN#=L)	RWS	Yes	0 (STRAP_NT_P2P_EN#=H) 1 (STRAP_NT_P2P_EN#=L)				
7	<b>Factory Test Only</b>	RWS	Yes	0				

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-64. 1DCh Debug Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
11:8	<b>Upstream Port ID</b> Upstream Port Number – Reads the external Strap value on the STRAP_UPSTRM_PORTSEL[3:0] inputs, at Reset de-assertion. When bit 15 ( <i>Hardware/Software Configuration Mode Control</i> ) is Cleared, software is not allowed to change this value.	RO	Yes		
	When bit 15 ( <i>Hardware/Software Configuration Mode Control</i> ) is Set, Upstream Port Number can be Set by software, using the values defined below.  <i>Note:</i> Port 0 is recommended for the upstream Port designation.  0h = Port 0 1h = Port 1 2h = Port 2 3h = Port 3 4h = Port 4 5h = Port 5 6h = Port 6 7h = Port 7 8h = Port 8 9h = Port 9 Ah = Port 10 Bh = Port 11 Ch = Port 12 Dh = Port 13 Eh = Port 14 Fh = Port 15	RW <sup>a</sup>	Yes	Set by STRAP_UPSTRM_PORTSEL[3:0] input levels, serial EEPROM, or I <sup>2</sup> C followed by a Soft Reset	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-64. 1DCh Debug Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
13:12	<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 8619 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>.</li> <li>When the PEX 8619 detects an internal Fatal error (ECC failure), the switch sends a Fatal Interrupt Message to the Host and asserts <b>FATAL_ERR#</b>. 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)</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>. This Fatal error detection blocks all 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 all the Mode 3 actions, the PEX 8619 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	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
14	<b>Factory Test Only</b>	RWS	Yes	0	
15	<p><b>Hardware/Software Configuration Mode Control</b> Allows software to configure which Port is the upstream Port, as well as which Port is a Legacy NT Port.</p> <p>0 = Upstream Port and NT Port selection by the <b>STRAP_UPSTRM_PORTSEL[3:0]</b> and <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs, respectively, which can be overridden by the serial EEPROM and/or I<sup>2</sup>C configuration mechanism. Cannot be changed by in-band software during runtime.</p> <p>1 = In-band software can change which Port is configured to be the upstream Port and NT Port, by writing new values to fields [11:8 and 27:24] (<b>Upstream Port ID</b> and <b>NT Port Number</b>, respectively), followed by issuance of a Hot Reset to the upstream Port. Bit 20 (<b>Upstream Port and NT-Link DL_Down Reset Propagation Disable</b>) must be Cleared.</p>	RWS	Yes	0	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABLE#=H)	NT Mode (STRAP_NT_ENABLE#=L)
16	<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 r2.0</i> <i>Note:</i> Only a Fundamental Reset serial EEPROM load affects this bit.	RWS	Yes	0	1
17	<b>Disable Serial EEPROM Load on Hot Reset</b> 0 = Enables serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state 1 = Disables serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state	RWS	Yes	0	
18	<b>NT Mode Enable</b> <i>Used only in NT mode.</i> NT mode (Intelligent Adapter) is enabled by the <a href="#">STRAP_NT_ENABLE#</a> input, which is overridden by the value of this bit if this register is programmed by the serial EEPROM upon Fundamental Reset. Software, serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state, and/or I <sup>2</sup> C are not allowed to change this value. 0 = NT mode is disabled (STRAP_NT_ENABLE#=H) 1 = NT mode is enabled (STRAP_NT_ENABLE#=L)	HwInit	Yes (Serial EEPROM only)	Set by STRAP_NT_ENABLE# input level or by serial EEPROM	
19	<b>NT Port DL_Down Reset Propagation Disable</b> <i>Used only in NT mode.</i> Setting this bit: <ul style="list-style-type: none"> <li>Enables the NT-Link Port to ignore a Hot Reset training sequence</li> <li>Blocks the NT-Link Port from manifesting an internal reset due to an NT-Link <i>DL_Down</i> event</li> </ul>	RWS	Yes	0	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
20	<b>Upstream Port and NT-Link DL_Down Reset Propagation Disable</b> Setting this bit: <ul style="list-style-type: none"> <li>Enables the upstream Port to ignore a Hot Reset training sequence,</li> <li>Blocks the PEX 8619 from manifesting an internal reset due to a DL_Down event, and</li> <li>Prevents the downstream Ports from issuing a Hot Reset to downstream devices when a Hot Reset or DL_Down event occurs on the upstream Link</li> </ul> If NT mode is enabled, Setting this bit additionally: <ul style="list-style-type: none"> <li>Enables the NT-Link Port to ignore a Hot Reset training sequence, and</li> <li>Blocks the PEX 8619 NT Port Link Interface from manifesting an internal reset due to a DL_Down event on the NT Port Link</li> </ul>	RWS	Yes		0
21	<b>Cut-Thru Enable</b> 0 = Disables Cut-Thru support 1 = Enables Cut-Thru support	RWS	Yes		1
22	<b>NT Security Mode Enable</b> Setting this bit blocks Memory-Mapped access from the NT-Link side to all PEX 8619 registers, except the NT Port Link Interface registers. For Memory Requests that target NT-Link <b>BAR0</b> offsets other than 1_1XXXh (68 to 72 KB) within the 128-KB range, Reads return the data value 0h, and Write data is ignored.	RWS	Yes		0
23	<b>Factory Test Only</b>	RWS	Yes		0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
27:24	<p><b>NT Port Number</b> <i>Used only in NT mode.</i></p> <p>When bit 18 (<i>NT Mode Enable</i>) is Set, and bit 15 (<i>Hardware/Software Configuration Mode Control</i>) is Cleared, the NT Port Number is Set by the <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs. This field is “Don’t Care” for Transparent mode. Software is not allowed to change this value.</p> <p><i>Used only in NT mode.</i></p> <p>When bits 18 (<i>NT Mode Enable</i>) and 15 (<i>Hardware/Software Configuration Mode Control</i>) are both Set, the NT Port Number selected by this field is Set by software, using the values defined below.</p> <p>0h = Port 0 1h = Port 1 2h = Port 2 3h = Port 3 4h = Port 4 5h = Port 5 6h = Port 6 7h = Port 7 8h = Port 8 9h = Port 9 Ah = Port 10 Bh = Port 11 Ch = Port 12 Dh = Port 13 Eh = Port 14 Fh = Port 15</p>	HwInit	Yes		

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
28	<p><b>Virtual Interface Access Enable</b> <i>Used only in NT mode.</i></p> <p>When the serial EEPROM is not present, the default value is 1; otherwise, the default value is 0.</p> <p>0 = Retries Type 0 Configuration TLP received on the NT Port Virtual Interface</p> <p>1 = Accepts Type 0 Configuration TLP on the NT Port Virtual Interface</p> <p><i>Notes:</i> This bit does not affect the PEX 8619 in Transparent mode, nor does it affect other transaction types.</p> <p><i>Set this bit to enable Configuration access to the NT Port Virtual Interface.</i></p>	RW	Yes		1
29	<p><b>Link Interface Access Enable</b> <i>Used only in NT mode.</i></p> <p>0 = Retries Type 0 Configuration Request received on the NT Port Link Interface</p> <p>1 = Accepts Type 0 Configuration Request received on the NT Port Link Interface</p> <p><i>Notes:</i> This bit does not affect the PEX 8619 in Transparent mode.</p> <p><i>Set this bit to enable Configuration access to the NT Port Link Interface.</i></p>	RW	Yes		0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-64. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
				Transparent Mode (STRAP_NT_ENABL E#=H)	NT Mode (STRAP_NT_ENABL E#=L)
30	<b>Inhibit EEPROM NT-Link Load on Hot Reset</b> <i>Used only in NT mode.</i> Inhibits serial EEPROM load of NT Port Link Interface registers when any one of the following conditions exist: <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [17:16] (<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 [20, 17:16] (<i>Upstream Port and NT-Link 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 17 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul>	RW	Yes	0	1
31	<b>Load Only EEPROM NT-Link on Hot Reset</b> <i>Used only in NT mode.</i> Load only serial EEPROM NT Port Link Interface register entries when any one of the following conditions exist: <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [17:16] (<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 [20, 17:16] (<i>Upstream Port and NT-Link 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 17 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul>	RW	Yes	0	1

a. Although these bits are RW, do not change by software.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-65. 1E0h 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 Counter</b> Traffic idle time to meet to enter the L0s Link PM state. 0 = Idle condition must last 1 µs 1 = Idle condition must last 4 µs		RW	Yes	0
1	<b>ASPM L1 Disable</b>		RW	Yes	0
2	<b>Not enabled</b> Functionality associated with this bit is enabled only on the downstream Ports.	Upstream	RW	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	RW	Yes	0
4:3	<b>Reserved</b>		RsvdP	No	00b
5	<b>Factory Test Only</b>		RW	Yes	0
6	<b>Reserved</b>		RsvdP	No	0
7	<b>Disable PCI PM L1 Entry</b>		RW	Yes	0
8	<b>DLLP Timeout Link Retrain Disable</b> Disable Link retraining when no Data Link Layer Packets (DLLPs) are received for more than 256 µs. 0 = Enables Link retraining when no DLLPs are received for more than 256 µs (default) 1 = DLLP Timeout is disabled		RW	Yes	0
9	<b>Factory Test Only</b>		RW	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		RW	Yes	0
11	<b>Factory Test Only</b>		RW	Yes	0
12	<b>NT Hot Plug Enable</b> Can be programmed only by serial EEPROM and/or I <sup>2</sup> C.		RO	Yes	0
14:13	<b>HP Parallel Port</b> <b>Not supported</b> No functionality, because Parallel Hot Plug is <b>not supported</b> .		RO	Yes	00b
15	<b>HPC GPIO Write in Progress</b> Indicates that the last Write operation to an I/O Expander <b>GPIO x Output Data</b> register (offsets 644h and 648h) is still in progress.		RO	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-65. 1E0h 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> Valid only for Port 0. All other Ports' values are unused. 0 = Enables Serial Hot Plug capability on all Ports 1 = Disables Serial Hot Plug capability on all Ports	0	RW	Yes	0
17	<b>40-Pin I/O Expander Enable</b> 0 = Enables 16-pin I/O Expanders for all downstream Ports that implement Serial Hot Plug 1 = Enables 40-pin I/O Expanders for all downstream Ports that implement Serial Hot Plug <i>Note:</i> Value of 1 can be enabled only by serial EEPROM (that is, neither software nor I <sup>2</sup> C can be used to enable the 40-Pin I/O Expander).	0	RW	Yes (See Note)	0
18	<b>HPC GPIO Config</b> Programs the GPIO direction for one GPIO. 0 = Input 1 = Output; register is RW		RW	Yes	0
19	<b>HPC GPIO Input/Output Value</b> If the external I <sup>2</sup> C I/O Expander GPIO pin is programmed as input ( <i>HPC GPIO Config</i> , bit 18, is Cleared), this bit reflects the logic value of the voltage on that Input pin. If the external I <sup>2</sup> C I/O Expander GPIO pin is programmed as output ( <i>HPC GPIO Config</i> , bit 18, is Set), the value written to this bit is written to the external GPIO pin, through the I <sup>2</sup> C Slave interface.		RW	Yes	0
20	<b>HPC I/O Reload</b> 1 = Serial Hot Plug Controller (I <sup>2</sup> C I/O Expander) Output pin values are re-loaded from field [26:21] ( <i>HPC Output Reload Value</i> ). After the action is complete, this bit is self-Clearing.		RW	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 re-loaded to the Hot Plug Controller outputs associated with the Port. Bit 21 = I/O Expander <b>PWRLED#</b> Bit 22 = I/O Expander <b>ATNLED#</b> Bit 23 = I/O Expander <b>PWREN</b> Bit 24 = I/O Expander <b>RECLKEN#</b> Bit 25 = I/O Expander <b>PERST#</b> Bit 26 = I/O Expander <b>INTERLOCK</b>		RW	Yes	0-0h
31:27	<b>Factory Test Only</b>		RW	Yes	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-66. 1E4h Egress Control and Status  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b>	RW	Yes	00b
2	<b>Factory Test Only</b>	RW	Yes	1
7:3	<b>Factory Test Only</b>	RW	Yes	0-0h
8	<b>Reader Header Prefetch Disable</b> 0 = Enables Reader Header Prefetch capability 1 = Disables Reader Header Prefetch capability	RW	Yes	0
9	<b>Vendor-Specific 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	RW	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 bit 11 ( <i>Egress Credit Timeout Value</i> ). Status is reflected in bits [19 and 16] ( <i>VCI EGR Credit Timeout Status</i> and <i>VCO EGR Credit Timeout Status</i> , respectively). 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.	RW	Yes	0
11	<b>Egress Credit Timeout Value</b> 0 = 384 to 512 ms 1 = 896 to 1,024 ms	RW	Yes	0
15:12	<b>Reserved</b>	RsvdP	No	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-66. 1E4h Egress Control and Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>VC0 EGR Credit Timeout Status</b> 0 = No timeout 1 = Timeout	RW1C	No	0
18:17	<b>VC0 EGR Credit Timeout Type</b> 00b = Posted 01b = Non-Posted 10b = Completion 11b = <i>Reserved</i>	RO	Yes	00b
19	<b>VC1 EGR Credit Timeout Status</b> 0 = No timeout 1 = Timeout	RW1C	No	0
21:20	<b>VC1 EGR Credit Timeout Type</b> 00b = Posted 01b = Non-Posted 10b = Completion 11b = <i>Reserved</i>	RO	Yes	00b
22	<b>Factory Test Only</b>	RW	Yes	1
23	<b>Factory Test Only</b>	RW	Yes	0
31:24	<b>Factory Test Only</b>	RW	Yes	AAh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-67. 1E8h Bad TLP Counter  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad TLP Counter</b> Counts the quantity of TLPs received with bad Link Cyclic Redundancy Check (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 14-68. 1EC<sub>H</sub> Bad DLLP Counter  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad DLLP Counter</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-69. 1F4h Software Lane Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Lane 0 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
1	<b>Lane 1 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
2	<b>Lane 2 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
3	<b>Lane 3 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
4	<b>Lane 4 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
5	<b>Lane 5 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
6	<b>Lane 6 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
7	<b>Lane 7 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-69. 1F4h Software Lane Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Lane 8 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
9	<b>Lane 9 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
10	<b>Lane 10 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
11	<b>Lane 11 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
12	<b>Lane 12 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
13	<b>Lane 13 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
14	<b>Lane 14 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
15	<b>Lane 15 Up Status</b> 0 = Lane is down 1 = Lane is up	RO	No	1
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-70. 1F8h 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 <a href="#">Current Link Speed</a> (offset 78h[19:16]), <a href="#">Negotiated Link Width</a> (offset 78h[25:20]), and <a href="#">Maximum Payload Size</a> (offset 70h[7:5]). x1 Link width = 255d x2 Link width = 217d x4 Link width = 118d x8 Link width = 107d	RWS	Yes	Set by STRAP_PORTCFG[3:0] input levels
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 register is a multiplier of the default internal timer values that are compliant to the <i>PCI Express Base r2.0</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

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

## 14.14.2 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 14-18](#) defines the register map.

Other Device-Specific PHY registers are detailed in:

- [Section 14.16.14, “Device-Specific Registers – Physical Layer \(Offsets B80h – C30h\)”](#)
- [Section 14.16.16, “Device-Specific Registers – Physical Layer \(Offsets E40h – EFCh\)”](#)

**Table 14-18. Device-Specific PHY Register Map  
(Offsets 200h – 25Ch) (Port 0, and also the NT Port Virtual  
Interface if Port 0 is a Legacy NT 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	
		Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status
		Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status
		<b>Factory Test Only</b>
		208h – 20Ch
		Physical Layer User Test Pattern, Bytes 0 through 3
		210h
		Physical Layer User Test Pattern, Bytes 4 through 7
		214h
		Physical Layer User Test Pattern, Bytes 8 through 11
		218h
		Physical Layer User Test Pattern, Bytes 12 through 15
		21Ch
		Even Ports – Physical Layer Command/Status
		220h
		Odd Ports – Physical Layer Command/Status
		224h
		Even Ports – Physical Layer Test
		228h
		Odd Ports – Physical Layer Test
		22Ch
<b>Reserved</b>		Even Ports – Disable/Quiet/Test Pattern Rate
<b>Reserved</b>		230h
<b>Reserved</b>		Odd Ports – Disable/Quiet/Test Pattern Rate
		234h
		Common Physical Layer Command/Status
<b>Reserved</b>		238h
<b>Reserved</b>		Common Physical Layer Safety and Test
		23Ch
		Even Ports – SerDes Quad 0 Diagnostic Data
		240h
		Odd Ports – SerDes Quad 2 Diagnostic Data
		244h
		Even Ports – SerDes Quad 1 Diagnostic Data
		248h
		Odd Ports – SerDes Quad 3 Diagnostic Data
		24Ch
		Even Ports – Physical Layer Safety and Test
		250h
		Odd Ports – Physical Layer Safety and Test
		254h
		Even Ports – User Test Pattern Delay
		258h
		Odd Ports – User Test Pattern Delay
		25Ch

**Notes:** In this section, the term “SerDes quad” or “quad” refers to assembling SerDes modules into groups of four contiguous Lanes for testing purposes.

[Table 14-19](#) defines the PEX 8619 Port and Lane configurations. Ports that are not configured nor enabled are invisible to software.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-19. Port Configurations**

<b>STRAP_PORTCFG [3:0]</b>	<b>Port 0</b>	<b>Port 1</b>	<b>Port 2</b>	<b>Port 3</b>	<b>Port 4</b>	<b>Port 5</b>	<b>Port 6</b>	<b>Port 7</b>	<b>Port 8</b>	<b>Port 9</b>	<b>Port 10</b>	<b>Port 11</b>	<b>Port 12</b>	<b>Port 13</b>	<b>Port 14</b>	<b>Port 15</b>
(default) 0h	x1	x1	x1	x1	x1	x1										
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	x1
2h	x4	x4	x1	x1							x1	x1	x1	x1	x1	x1
3h	x4	x4	x4	x1								x1		x1	x1	x1
4h	x4	x4	x4	x4												
5h	x8	x1		x1		x1		x1								
6h	x8	x4		x1							x1		x1		x1	x1
7h	x8	x4		x4												
8h	x8	x8														

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register is used for specifying the pre-determined quantity of Lanes that detected a Receiver during a LTSSM <i>Detect</i> state, but never detected an exit from Electrical Idle. Because the PEX 8619 has multiple Port configurations, a Mask register is used, rather than specifying a number. 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> state.				
<i>Note:</i> Masking Electrical Idle detect does not affect the inferred Electrical Idle detection. Use the SerDes x Mask Electrical Idle Detect bits with caution.				
0	<b>Electrical Idle on SerDes 0 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
1	<b>Electrical Idle on SerDes 1 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
2	<b>Electrical Idle on SerDes 2 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
3	<b>Electrical Idle on SerDes 3 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Electrical Idle on SerDes 4 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition. 1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
5	<b>Electrical Idle on SerDes 5 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition. 1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
6	<b>Electrical Idle on SerDes 6 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition. 1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
7	<b>Electrical Idle on SerDes 7 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition. 1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>SerDes 0 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
9	<b>SerDes 1 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
10	<b>SerDes 2 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
11	<b>SerDes 3 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>SerDes 4 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
13	<b>SerDes 5 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
14	<b>SerDes 6 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
15	<b>SerDes 7 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>SerDes 0 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
17	<b>SerDes 1 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
18	<b>SerDes 2 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
19	<b>SerDes 3 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
20	<b>SerDes 4 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
21	<b>SerDes 5 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
22	<b>SerDes 6 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
23	<b>SerDes 7 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver.  The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-71. 200h Even Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Receiver Detected on Lane 0</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 0.	RO	No	Set by SerDes
25	<b>Receiver Detected on Lane 1</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 1.	RO	No	Set by SerDes
26	<b>Receiver Detected on Lane 2</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 2.	RO	No	Set by SerDes
27	<b>Receiver Detected on Lane 3</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 3.	RO	No	Set by SerDes
28	<b>Receiver Detected on Lane 4</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 4.	RO	No	Set by SerDes
29	<b>Receiver Detected on Lane 5</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 5.	RO	No	Set by SerDes
30	<b>Receiver Detected on Lane 6</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 6.	RO	No	Set by SerDes
31	<b>Receiver Detected on Lane 7</b> Returns the Receiver's LTSSM <i>Detect</i> state status, and reads back as 1 when a Receiver is detected on Lane 7.	RO	No	Set by SerDes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register is used for specifying the pre-determined quantity of Lanes that detected a Receiver during a LTSSM <i>Detect</i> state, but never detected an exit from Electrical Idle. Because the PEX 8619 has multiple Port configurations, a Mask register is used, rather than specifying a number. 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> state.				
<i>Note:</i> Masking Electrical Idle detect does not affect the inferred Electrical Idle detection. Use the SerDes x Mask Electrical Idle Detect bits with caution.				
0	<b>Electrical Idle on SerDes 8 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
1	<b>Electrical Idle on SerDes 9 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
2	<b>Electrical Idle on SerDes 10 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
3	<b>Electrical Idle on SerDes 11 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Electrical Idle on SerDes 12 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
5	<b>Electrical Idle on SerDes 13 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
6	<b>Electrical Idle on SerDes 14 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1
7	<b>Electrical Idle on SerDes 15 Causes Entry to Compliance State</b> When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> state cannot be entered, due to the Electrical Idle condition.  1 = Corresponding 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> state, to cause entry to the <i>Polling.Compliance</i> state	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>SerDes 8 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
9	<b>SerDes 9 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
10	<b>SerDes 10 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
11	<b>SerDes 11 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>SerDes 12 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
13	<b>SerDes 13 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
14	<b>SerDes 14 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
15	<b>SerDes 15 Mask Electrical Idle Detect</b> 0 = Analog Electrical Idle detection is enabled for this SerDes. 1 = Analog Electrical Idle detection is disabled for this SerDes. Electrical Idle entrance inference, however, can be enabled by one or more available methods only if this bit is Set, and the Port's <b>SerDes Control</b> register (offset C28h) <i>Port x Electrical Idle Inference Disable</i> bit associated with this SerDes is Cleared.	RWS	Yes	0
16	<b>SerDes 8 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
17	<b>SerDes 9 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
18	<b>SerDes 10 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
19	<b>SerDes 11 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-72. 204h Odd Port Lanes – Electrical Idle for Compliance Mask, and Receiver Detect Mask and Status (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>SerDes 12 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
21	<b>SerDes 13 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
22	<b>SerDes 14 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
23	<b>SerDes 15 Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes. The corresponding Lane will always detect a Receiver. The PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.	RWS	Yes	0
24	<b>Receiver Detected on Lane 8</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 8.	RO	No	Set by SerDes
25	<b>Receiver Detected on Lane 9</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 9.	RO	No	Set by SerDes
26	<b>Receiver Detected on Lane 10</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 10.	RO	No	Set by SerDes
27	<b>Receiver Detected on Lane 11</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 11.	RO	No	Set by SerDes
28	<b>Receiver Detected on Lane 12</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 12.	RO	No	Set by SerDes
29	<b>Receiver Detected on Lane 13</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 13.	RO	No	Set by SerDes
30	<b>Receiver Detected on Lane 14</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 14.	RO	No	Set by SerDes
31	<b>Receiver Detected on Lane 15</b> Returns the Receiver's LTSSM Detect state status, and reads back as 1 when a Receiver is detected on Lane 15.	RO	No	Set by SerDes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-73. 210h Physical Layer User Test Pattern, Bytes 0 through 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
UTP Bytes 0 through 3. Used for Digital Far-End Loopback testing.				
<i>Note:</i> A 16-byte test pattern can be written to register offsets 210h through 21Ch. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 210h is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 21Ch is transmitted last. (Refer to <a href="#">Section 20.2.3, “Digital Loopback Master Mode,”</a> for further details.) Every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified.				
7:0	Byte 0 of the UTP. This is the first byte transferred.	RW	Yes	00h
15:8	Byte 1 of the UTP.	RW	Yes	00h
23:16	Byte 2 of the UTP.	RW	Yes	00h
31:24	Byte 3 of the UTP.	RW	Yes	00h

**Register 14-74. 214h Physical Layer User Test Pattern, Bytes 4 through 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> A 16-byte test pattern can be written to register offsets 210h through 21Ch. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 210h is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 21Ch is transmitted last. (Refer to <a href="#">Section 20.2.3, “Digital Loopback Master Mode,”</a> for further details.) Every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified.				
7:0	Byte 4 of the UTP. This is the fifth byte transferred.	RW	Yes	00h
15:8	Byte 5 of the UTP.	RW	Yes	00h
23:16	Byte 6 of the UTP.	RW	Yes	00h
31:24	Byte 7 of the UTP.	RW	Yes	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-75. 218h Physical Layer User Test Pattern, Bytes 8 through 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> A 16-byte test pattern can be written to register offsets 210h through 21Ch. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 210h is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 21Ch is transmitted last. (Refer to Section 20.2.3, “Digital Loopback Master Mode,” for further details.) Every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified.				
7:0	Byte 8 of the UTP. This is the ninth byte transferred.	RW	Yes	00h
15:8	Byte 9 of the UTP.	RW	Yes	00h
23:16	Byte 10 of the UTP.	RW	Yes	00h
31:24	Byte 11 of the UTP.	RW	Yes	00h

**Register 14-76. 21Ch Physical Layer User Test Pattern, Bytes 12 through 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> A 16-byte test pattern can be written to register offsets 210h through 21Ch. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 210h is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 21Ch is transmitted last. (Refer to Section 20.2.3, “Digital Loopback Master Mode,” for further details.) Every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified.				
7:0	Byte 12 of the UTP. This is the thirteenth byte transferred.	RW	Yes	00h
15:8	Byte 13 of the UTP.	RW	Yes	00h
23:16	Byte 14 of the UTP.	RW	Yes	00h
31:24	Byte 15 of the UTP.	RW	Yes	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides the Loopback, Scrambler Disable, and Compliance Receive commands, and Ready as Loopback Master status, for each even-numbered Port.				
0	<b>Port 0 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
1	<b>Port 0 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
2	<b>Port 0 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> state 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> state	RW	Yes	0
3	<b>Port 0 Ready as Loopback Master</b> Link Training and Status State Machine (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> state as a Loopback Master. The LTSSM remains in this state, until bit 0 ( <i>Port 0 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 2 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
5	<b>Port 2 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
6	<b>Port 2 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> state 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> state	RW	Yes	0
7	<b>Port 2 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> state as a Loopback Master. The LTSSM remains in this state, until bit 4 ( <i>Port 2 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 4 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.	RW	Yes	0
9	<b>Port 4 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
10	<b>Port 4 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> state 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> state	RW	Yes	0
11	<b>Port 4 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> state as a Loopback Master. The LTSSM remains in this state, until bit 8 ( <i>Port 4 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 6 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
13	<b>Port 6 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
14	<b>Port 6 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> state 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> state	RW	Yes	0
15	<b>Port 6 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> state as a Loopback Master. The LTSSM remains in this state, until bit 12 ( <i>Port 6 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 8 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.	RW	Yes	0
17	<b>Port 8 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
18	<b>Port 8 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> state 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> state	RW	Yes	0
19	<b>Port 8 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> state as a Loopback Master. The LTSSM remains in this state, until bit 16 ( <i>Port 8 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Port 10 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
21	<b>Port 10 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
22	<b>Port 10 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> state 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> state	RW	Yes	0
23	<b>Port 10 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> state as a Loopback Master. The LTSSM remains in this state, until bit 20 ( <i>Port 10 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Port 12 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.	RW	Yes	0
25	<b>Port 12 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
26	<b>Port 12 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> state 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> state	RW	Yes	0
27	<b>Port 12 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> state as a Loopback Master. The LTSSM remains in this state, until bit 24 ( <i>Port 12 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-77. 220h Even Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Port 14 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
29	<b>Port 14 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
30	<b>Port 14 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> state 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> state	RW	Yes	0
31	<b>Port 14 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> state as a Loopback Master. The LTSSM remains in this state, until bit 28 ( <i>Port 14 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides the Loopback, Scrambler Disable, and Compliance Receive commands, and Ready as Loopback Master status, for each odd-numbered Port.				
0	<b>Port 1 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.	RW	Yes	0
1	<b>Port 1 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
2	<b>Port 1 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> state 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> state	RW	Yes	0
3	<b>Port 1 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> state as a Loopback Master. The LTSSM remains in this state, until bit 0 ( <i>Port 1 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 3 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
5	<b>Port 3 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
6	<b>Port 3 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> state 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> state	RW	Yes	0
7	<b>Port 3 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> state as a Loopback Master. The LTSSM remains in this state, until bit 4 ( <i>Port 3 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 5 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.	RW	Yes	0
9	<b>Port 5 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
10	<b>Port 5 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> state 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> state	RW	Yes	0
11	<b>Port 5 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> state as a Loopback Master. The LTSSM remains in this state, until bit 8 ( <i>Port 5 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 7 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
13	<b>Port 7 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
14	<b>Port 7 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> state 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> state	RW	Yes	0
15	<b>Port 7 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> state as a Loopback Master. The LTSSM remains in this state, until bit 12 ( <i>Port 7 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 9 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.	RW	Yes	0
17	<b>Port 9 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
18	<b>Port 9 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> state 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> state	RW	Yes	0
19	<b>Port 9 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> state as a Loopback Master. The LTSSM remains in this state, until bit 16 ( <i>Port 9 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Port 11 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
21	<b>Port 11 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
22	<b>Port 11 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> state 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> state	RW	Yes	0
23	<b>Port 11 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> state as a Loopback Master. The LTSSM remains in this state, until bit 20 ( <i>Port 11 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Port 13 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.	RW	Yes	0
25	<b>Port 13 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
26	<b>Port 13 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> state 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> state	RW	Yes	0
27	<b>Port 13 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> state as a Loopback Master. The LTSSM remains in this state, until bit 24 ( <i>Port 13 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-78. 224h Odd Ports – Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Port 15 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 to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.	RW	Yes	0
29	<b>Port 15 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane, and causes the <i>Disable Scrambling</i> Training Control Bit to be Set in the transmitted Training Sets. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> state. 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	RW	Yes	0
30	<b>Port 15 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> state 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> state	RW	Yes	0
31	<b>Port 15 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> state as a Loopback Master. The LTSSM remains in this state, until bit 28 ( <i>Port 15 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8619 exits the <i>Loopback.Active</i> state.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides controls to enable various PHY test modes, for the Even Ports.				
0	<b>Port 0 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	RW	Yes	0
1	<b>Port 2 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	RW	Yes	0
2	<b>Port 4 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	RW	Yes	0
3	<b>Port 6 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	RW	Yes	0
4	<b>Port 8 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	RW	Yes	0
5	<b>Port 10 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	RW	Yes	0
6	<b>Port 12 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	RW	Yes	0
7	<b>Port 14 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	RW	Yes	0
10:8	<b>Factory Test Only</b>	RW	Yes	000b
15, 11	<b>Detect.Quiet Wait Time Select Code[1:0]</b> Selects the amount of time to wait during the <i>Detect.Quiet</i> state, 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. 00b = 0 ms 01b = 4 ms 10b = 8 ms 11b = 12 ms	RWS	Yes	00b
14:12	<b>Factory Test Only</b>	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Lane 0 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
17	<b>Lane 1 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
18	<b>Lane 2 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
19	<b>Lane 3 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
20	<b>Lane 4 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
21	<b>Lane 5 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
22	<b>Lane 6 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
23	<b>Lane 7 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>Lane 0 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
25	<p><b>Lane 1 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26	<p><b>Lane 2 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><i>Note:</i> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
27	<p><b>Lane 3 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><i>Note:</i> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>Lane 4 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
29	<p><b>Lane 5 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-79. 228h Even Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30	<p><b>Lane 6 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><i>Note:</i> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
31	<p><b>Lane 7 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><i>Note:</i> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides controls to enable various PHY test modes, for the Odd Ports.				
0	<b>Port 1 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	RW	Yes	0
1	<b>Port 3 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	RW	Yes	0
2	<b>Port 5 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	RW	Yes	0
3	<b>Port 7 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	RW	Yes	0
4	<b>Port 9 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	RW	Yes	0
5	<b>Port 11 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	RW	Yes	0
6	<b>Port 13 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	RW	Yes	0
7	<b>Port 15 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	RW	Yes	0
10:8	<b>Factory Test Only</b>	RW	Yes	000b
15, 11	<b>Detect.Quiet Wait Time Select Code[1:0]</b> Selects the amount of time to wait during the <i>Detect.Quiet</i> state, 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. 00b = 0 ms 01b = 4 ms 10b = 8 ms 11b = 12 ms	RWS	Yes	00b
14:12	<b>Factory Test Only</b>	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Lane 8 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
17	<b>Lane 9 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
18	<b>Lane 10 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
19	<b>Lane 11 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
20	<b>Lane 12 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
21	<b>Lane 13 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
22	<b>Lane 14 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0
23	<b>Lane 15 Serial Loopback Path Enable</b> 1 = Lane enables the Serial <i>Loopback</i> (Master) path, regardless of the LTSSM state	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>Lane 8 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
25	<p><b>Lane 9 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26	<p><b>Lane 10 Parallel Loopback Path Enable</b>  The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.  It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.  0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane  1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state  <b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
27	<p><b>Lane 11 Parallel Loopback Path Enable</b>  The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.  It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.  0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane  1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state  <b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>Lane 12 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
29	<p><b>Lane 13 Parallel Loopback Path Enable</b></p> <p>The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.</p> <p>It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.</p> <p>0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane 1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state</p> <p><b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-80. 22Ch Odd Ports – Physical Layer Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30	<p><b>Lane 14 Parallel Loopback Path Enable</b>  The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.  It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.  0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane  1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state  <b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0
31	<p><b>Lane 15 Parallel Loopback Path Enable</b>  The <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) overrides this control, and <b>must be Cleared</b> when using Parallel “Digital” <i>Loopback</i>.  It is recommended that the Port associated with this Lane be placed into a <i>Port Disable</i> state, by Setting the Port’s <b>Even/Odd Port Disable</b> register <i>Disable Port x</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0]), followed by a <i>Port Quiet</i> state, by Setting the Port’s <i>Hold Port x Quiet</i> bit (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]), before Setting this bit.  0 = Parallel “Digital” <i>Loopback</i> (Slave) path is disabled for this Lane  1 = Lane manually enables the Parallel “Digital” <i>Loopback</i> (Slave) path, regardless of the LTSSM state  <b>Note:</b> This path is automatically enabled when the LTSSM enters the <i>Loopback.Active</i> state, as a <i>Loopback Slave</i>.</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register is used to disable or enable the LTSSM within individual Even Ports. The bits are intended to be used in lieu of placing the Port into the <i>Loopback.Active</i> state as a Loopback Master. These bits enable the test patterns to be transmitted, with or without a device attached at the far end. The recommended usage is as follows:</p> <ol style="list-style-type: none"> <li>1. Set the Port's <i>Disable Port x</i> and <i>Port x Quiet</i> bits.             <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> state. If no device is attached, it is not necessary to Set the Port's <i>Disable Port x</i> bit.</li> <li>• If 5.0 GT/s is needed, also Set the Port's <i>Test Pattern x Rate</i> bit.</li> </ul> </li> <li>2. If Set, Clear the Port's <i>Disable Port x</i> bit.</li> <li>3. Load the UTP registers and enable UTP transmission, or just enable PRBS transmission.</li> </ol>				
0	<p><b>Disable Port 0</b></p> <p>While Port 0 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 0.</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 0, if it is currently in, or returns to, that state. Unconditionally disables Port 0. This is different from the Link Training and Status State Machine (LTSSM) <i>Disabled</i> state, in that Port 0 does not attempt to enter this state. If Port 0 is idle, it ceases attempting to detect a Receiver. If Port 0 is up, it immediately returns to the <i>Detect.Quiet</i> state 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 the LTSSM <i>Detect</i> state. Port 0 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
1	<p><b>Disable Port 2</b></p> <p>While Port 2 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 2.</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 2, if it is currently in, or returns to, that state. Unconditionally disables Port 2. This is different from the LTSSM <i>Disabled</i> state, in that Port 2 does not attempt to enter this state. If Port 2 is idle, it ceases attempting to detect a Receiver. If Port 2 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 2 remains disabled until this bit is Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<p><b>Disable Port 4</b></p> <p>While Port 4 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 4. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 4, if it is currently in, or returns to, that state. Unconditionally disables Port 4. This is different from the LTSSM <i>Disabled</i> state, in that Port 4 does not attempt to enter this state. If Port 4 is idle, it ceases attempting to detect a Receiver. If Port 4 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 4 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
3	<p><b>Disable Port 6</b></p> <p>While Port 6 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 6. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 6, if it is currently in, or returns to, that state. Unconditionally disables Port 6. This is different from the LTSSM <i>Disabled</i> state, in that Port 6 does not attempt to enter this state. If Port 6 is idle, it ceases attempting to detect a Receiver. If Port 6 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 6 remains disabled until this bit is Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>Disable Port 8</b></p> <p>While Port 8 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 8. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 8, if it is currently in, or returns to, that state. Unconditionally disables Port 8. This is different from the LTSSM <i>Disabled</i> state, in that Port 8 does not attempt to enter this state. If Port 8 is idle, it ceases attempting to detect a Receiver. If Port 8 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 8 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
5	<p><b>Disable Port 10</b></p> <p>While Port 10 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 10. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 10, if it is currently in, or returns to, that state. Unconditionally disables Port 10. This is different from the LTSSM <i>Disabled</i> state, in that Port 10 does not attempt to enter this state. If Port 10 is idle, it ceases attempting to detect a Receiver. If Port 10 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 10 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
6	<p><b>Disable Port 12</b></p> <p>While Port 12 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 12. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 12, if it is currently in, or returns to, that state. Unconditionally disables Port 12. This is different from the LTSSM <i>Disabled</i> state, in that Port 12 does not attempt to enter this state. If Port 12 is idle, it ceases attempting to detect a Receiver. If Port 12 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 12 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
7	<p><b>Disable Port 14</b></p> <p>While Port 14 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 14. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 14, if it is currently in, or returns to, that state. Unconditionally disables Port 14. This is different from the LTSSM <i>Disabled</i> state, in that Port 14 does not attempt to enter this state. If Port 14 is idle, it ceases attempting to detect a Receiver. If Port 14 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 14 remains disabled until this bit is Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<p><b>Hold Port 0 Quiet</b></p> <p>Unlike bit 0 (<i>Disable Port 0</i>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 0 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 0 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
9	<p><b>Hold Port 2 Quiet</b></p> <p>Unlike bit 1 (<i>Disable Port 2</i>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 2 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 2 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
10	<p><b>Hold Port 4 Quiet</b></p> <p>Unlike bit 2 (<i>Disable Port 4</i>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 4 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 4 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
11	<p><b>Hold Port 6 Quiet</b></p> <p>Unlike bit 3 (<i>Disable Port 6</i>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 6 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 6 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<p><b>Hold Port 8 Quiet</b></p> <p>Unlike bit 4 (<a href="#">Disable Port 8</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 8 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 8 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
13	<p><b>Hold Port 10 Quiet</b></p> <p>Unlike bit 5 (<a href="#">Disable Port 10</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 10 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 10 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
14	<p><b>Hold Port 12 Quiet</b></p> <p>Unlike bit 6 (<a href="#">Disable Port 12</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 12 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 12 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
15	<p><b>Hold Port 14 Quiet</b></p> <p>Unlike bit 7 (<a href="#">Disable Port 14</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 14 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 14 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 0 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 8 ( <i>Hold Port 0 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 0 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
17	<b>Port 2 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 9 ( <i>Hold Port 2 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 2 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
18	<b>Port 4 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 10 ( <i>Hold Port 4 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 4 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
19	<b>Port 6 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 11 ( <i>Hold Port 6 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 6 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-81. 230h Even Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Port 8 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 12 ( <i>Hold Port 8 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 8 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
21	<b>Port 10 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 13 ( <i>Hold Port 10 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 10 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
22	<b>Port 12 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 14 ( <i>Hold Port 12 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 12 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
23	<b>Port 14 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 15 ( <i>Hold Port 14 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 14 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register is used to disable or enable the LTSSM within individual Odd Ports. The bits are intended to be used in lieu of placing the Port into the <i>Loopback.Active</i> state as a Loopback Master. These bits enable the test patterns to be transmitted, with or without a device attached at the far end. The recommended usage is as follows:				
<ol style="list-style-type: none"> <li>1. Set the Port's <i>Disable Port x</i> and <i>Port x Quiet</i> bits.             <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> state. If no device is attached, it is not necessary to Set the Port's <i>Disable Port x</i> bit.</li> <li>• If 5.0 GT/s is needed, also Set the Port's <i>Test Pattern x Rate</i> bit.</li> </ul> </li> <li>2. If Set, Clear the Port's <i>Disable Port x</i> bit.</li> <li>3. Load the UTP registers and enable UTP transmission, or just enable PRBS transmission.</li> </ol>				
0	<p><b>Disable Port 1</b></p> <p>While Port 1 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 1. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 1, if it is currently in, or returns to, that state. Unconditionally disables Port 1. This is different from the LTSSM <i>Disabled</i> state, in that Port 1 does not attempt to enter this state. If Port 1 is idle, it ceases attempting to detect a Receiver. If Port 1 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 1 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
1	<p><b>Disable Port 3</b></p> <p>While Port 3 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 3. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 3, if it is currently in, or returns to, that state. Unconditionally disables Port 3. This is different from the LTSSM <i>Disabled</i> state, in that Port 3 does not attempt to enter this state. If Port 3 is idle, it ceases attempting to detect a Receiver. If Port 3 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 3 remains disabled until this bit is Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<p><b>Disable Port 5</b></p> <p>While Port 5 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 5.</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 5, if it is currently in, or returns to, that state. Unconditionally disables Port 5. This is different from the LTSSM <i>Disabled</i> state, in that Port 5 does not attempt to enter this state. If Port 5 is idle, it ceases attempting to detect a Receiver. If Port 5 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 5 remains disabled until this bit is Cleared.</p>	RWS	Yes	0
3	<p><b>Disable Port 7</b></p> <p>While Port 7 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state.</p> <p>0 = Enables Link Training operation on Port 7.</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 7, if it is currently in, or returns to, that state. Unconditionally disables Port 7. This is different from the LTSSM <i>Disabled</i> state, in that Port 7 does not attempt to enter this state. If Port 7 is idle, it ceases attempting to detect a Receiver. If Port 7 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 7 remains disabled until this bit is Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Disable Port 9</b> While Port 9 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state. 0 = Enables Link Training operation on Port 9. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 9, if it is currently in, or returns to, that state. Unconditionally disables Port 9. This is different from the LTSSM <i>Disabled</i> state, in that Port 9 does not attempt to enter this state. If Port 9 is idle, it ceases attempting to detect a Receiver. If Port 9 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 9 remains disabled until this bit is Cleared.	RWS	Yes	0
5	<b>Disable Port 11</b> While Port 11 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state. 0 = Enables Link Training operation on Port 11. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 11, if it is currently in, or returns to, that state. Unconditionally disables Port 11. This is different from the LTSSM <i>Disabled</i> state, in that Port 11 does not attempt to enter this state. If Port 11 is idle, it ceases attempting to detect a Receiver. If Port 11 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 11 remains disabled until this bit is Cleared.	RWS	Yes	0
6	<b>Disable Port 13</b> While Port 13 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state. 0 = Enables Link Training operation on Port 13. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 13, if it is currently in, or returns to, that state. Unconditionally disables Port 13. This is different from the LTSSM <i>Disabled</i> state, in that Port 13 does not attempt to enter this state. If Port 13 is idle, it ceases attempting to detect a Receiver. If Port 13 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 13 remains disabled until this bit is Cleared.	RWS	Yes	0
7	<b>Disable Port 15</b> While Port 15 is disabled, Receiver termination is disabled and the SerDes that belong to the Port are placed into the L1 Link PM state. 0 = Enables Link Training operation on Port 15. 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 15, if it is currently in, or returns to, that state. Unconditionally disables Port 15. This is different from the LTSSM <i>Disabled</i> state, in that Port 15 does not attempt to enter this state. If Port 15 is idle, it ceases attempting to detect a Receiver. If Port 15 is up, it immediately returns to the <i>Detect.Quiet</i> state and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to the LTSSM <i>Detect</i> state. Port 15 remains disabled until this bit is Cleared.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<p><b>Hold Port 1 Quiet</b></p> <p>Unlike bit 0 (<a href="#">Disable Port 1</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 1 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 1 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
9	<p><b>Hold Port 3 Quiet</b></p> <p>Unlike bit 1 (<a href="#">Disable Port 3</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 3 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 3 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
10	<p><b>Hold Port 5 Quiet</b></p> <p>Unlike bit 2 (<a href="#">Disable Port 5</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 5 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 5 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
11	<p><b>Hold Port 7 Quiet</b></p> <p>Unlike bit 3 (<a href="#">Disable Port 7</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 7 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM)</p> <p>1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 7 if it is currently in, or returns to, that state</p> <p><b>Note:</b> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<p><b>Hold Port 9 Quiet</b></p> <p>Unlike bit 4 (<a href="#">Disable Port 9</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 9 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 9 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
13	<p><b>Hold Port 11 Quiet</b></p> <p>Unlike bit 5 (<a href="#">Disable Port 11</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 11 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 11 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
14	<p><b>Hold Port 13 Quiet</b></p> <p>Unlike bit 6 (<a href="#">Disable Port 13</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 13 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 13 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0
15	<p><b>Hold Port 15 Quiet</b></p> <p>Unlike bit 7 (<a href="#">Disable Port 15</a>), this bit does not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. Port 15 can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> state.</p> <p>0 = LTSSM is allowed to exit the <i>Detect.Quiet</i> state in a normal manner (no effect on the LTSSM) 1 = LTSSM remains in the <i>Detect.Quiet</i> state on Port 15 if it is currently in, or returns to, that state</p> <p><i>Note:</i> Use this bit when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> state as a Loopback Master.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 1 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 8 ( <i>Hold Port 1 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 1 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
17	<b>Port 3 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 9 ( <i>Hold Port 3 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 3 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
18	<b>Port 5 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 10 ( <i>Hold Port 5 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 5 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
19	<b>Port 7 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 11 ( <i>Hold Port 7 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 7 Quiet</i> bit is Set). 0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-82. 234h Odd Ports – Disable/Quiet/Test Pattern Rate  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Port 9 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 12 ( <i>Hold Port 9 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 9 Quiet</i> bit is Set).  0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
21	<b>Port 11 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 13 ( <i>Hold Port 11 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 11 Quiet</i> bit is Set).  0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
22	<b>Port 13 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 14 ( <i>Hold Port 13 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 13 Quiet</i> bit is Set).  0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
23	<b>Port 15 Test Pattern x Rate</b> Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if bit 15 ( <i>Hold Port 15 Quiet</i> ) is also Set (manual rate selection is enabled only when the <i>Hold Port 15 Quiet</i> bit is Set).  0 = UTP is transmitted at 2.5 GT/s 1 = UTP is transmitted at 5.0 GT/s	RWS	Yes	0
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-83. 238h Common Physical Layer Command/Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:0	<b>Number of Ports Available</b> Returns the quantity of enabled Ports, based upon the selected Port configuration.	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
5	<b>Upstream Cross-Link Enable</b> 0 = Disables upstream cross-link, upstream Port cannot be connected to another upstream Port 1 = Enables upstream cross-link, upstream Port can be connected to another upstream Port	RWS	Yes	1
6	<b>Downstream Cross-Link Enable</b> 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	RWS	Yes	1
7	<b>Elastic Buffer Low-Latency Mode Disable</b> 0 = Enables Elastic Buffer Low-Latency mode 1 = Disables Elastic Buffer Low-Latency mode	RWS	Yes	0
15:8	<b>N_FTS Value</b> Number of Fast Training Sets (N_FTS) value to transmit (in Training Sets).	RWS	Yes	64h
31:16	<b>User Test Pattern K-Code Flag</b> The corresponding UTP byte is transmitted as a kcode. <i><b>Caution:</b> Use caution when turning on k-characters, because the transmit logic does not examine illegal codes for validity. Also, sequences of control codes that can be detected as legal SKIP Ordered-Sets in the middle of the data pattern can confuse the Receive data checking logic. Therefore, it is recommended to not turn on k-characters when testing with a UTP.</i>	RWS	Yes	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-84. 23Ch Common Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register contains the PHY Safety and Test Control bits that are common to all PEX 8619 Ports and Lanes.				
0	<b>L0 to Recovery on Unexpected Electrical Idle Disable</b> 0 = LTSSM transitions from the L0 Link PM state to the <i>Recovery</i> state when an Electrical Idle is detected, without first receiving an Electrical Idle Ordered-Set 1 = LTSSM does not transition from the L0 Link PM state to the <i>Recovery</i> state when an Electrical Idle is detected, without first receiving an Electrical Idle Ordered-Set	RWS	Yes	0
1	<b>Framer Filter K28 Disable</b> Enabling the K28 Framer Filter increases robustness of the link by not retraining the Link upon detection of a single COM symbol. 0 = Four consecutive COM symbols must be detected within a packet, to cause the Link to be retrained 1 = If a COM symbol is detected within a packet, loss of symbol lock is assumed, and the Link is immediately retrained	RWS	Yes	0
3:2	<b>Factory Test Only</b>	RsvdP	No	00b
4	<b>Retrain on Receive Error Burst</b> 1 = LTSSM enters the <i>Recovery</i> state, if 256 Receiver errors are detected within 1 ms	RWS	Yes	0
5	<b>Lane Reversal Disable</b> 0 = Enables Lane reversal support on all Ports 1 = Disables Lane reversal support on all Ports	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
7	<b>Reserved</b>	RsvdP	No	0
19:8	<b>SKIP Ordered-Set Interval</b> Specifies the SKIP Ordered-Set interval (in symbol times). When a value of 000h is written, SKIP Ordered-Set transmission is disabled. 000h = Disables SKIP Ordered-Set transmission 49Ch = Minimum interval (1,180 symbol times) 602h = Maximum interval (1,538 symbol times) <i>Note:</i> A high value (such as FFFh) can cause the Link to fail.	RWS	Yes	49Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-84. 23Ch Common Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	RsvdP	No	0
21	<p><b>Analog Loopback Enable</b> 0 = PEX 8619 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 8619 then loops back data through the Elastic buffer, 8b/10b decoder, and 8b/10b encoder.</p> <p>1 = Loopback point of all Ports is before the Elastic buffer. This means that data recovered from the Serial data in the recovered Receive Clock domain is re-serialized, then re-transmitted in that same recovered clock domain. This allows the Loopback Master to transmit and receive a user test pattern (UTP) in an asynchronous clocking system. It is also the <b>required</b> mode for re-transmitting a PRBS pattern back to the Loopback Master. Overrides the Lane's Parallel "Digital" Loopback Setting (<b>Even/Odd Port Physical Test</b> register <i>Lane x Parallel Loopback Path Enable</i> bits – Even Ports, offset 228h[31:24], and Odd Ports, offset 22Ch[31:24]).</p> <p><i>Note:</i> This bit must be Cleared when using Parallel "Digital" Loopback.</p>	RWS	Yes	0
22	<b>Reserved</b>	RsvdP	No	0
23	<b>Factory Test Only</b>	RW	Yes	0
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-85. 240h Even Ports – SerDes Quad 0 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
There are four <b>Even/Odd Port SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <b>240h</b> through <b>24Ch</b> , in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port. 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]).				
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. 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.				
This register is used to retrieve Diagnostic Test results for SerDes[0-3].				
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.	RO	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.	RO	No	00h
23:16	<b>UTP/PRBS Error Counter</b> Receiver Detected flags. Returns the quantity of errors detected by the UTP (bit 30 is Cleared) or PRBS (bit 30 is Set) Data Checkers. The Error Counter saturates at 255, and is Cleared when UTP/PRBS is disabled. <b>UTP Mode</b> To Clear the Counter, disable UTP mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x User Test Pattern Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>258h</b> [23:16], and Odd Ports, offset <b>25Ch</b> [23:16]). <b>PRBS Mode</b> To Clear the Counter, disable PRBS mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x PRBS Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[31:24], and Odd Ports, offset 25Ch[31:24]).	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-85. 240h Even Ports – SerDes Quad 0 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<p><b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[0-3]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [15:0] 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] are selected with corresponding binary codes from 0-3.</p> <p><i>Note:</i> To obtain diagnostic data on all SerDes in the quad, run a test, then cycle these bits.</p>	RW	Yes	00b
29:26	<p><b>PRBS Error Injection</b> 1h = Causes the pattern generators for this SerDes quad's Lanes to insert an error into the transmitted pattern, once every 127 words</p>	RW	No	0h
30	<p><b>PRBS Counter-/UTP Counter</b> 0 = Indicates that field [23:16] (UTP/PRBS Error Counter) is the UTP Error Counter 1 = Indicates that field [23:16] (UTP/PRBS Error Counter) is the PRBS Error Counter</p>	RO	No	0
31	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-86. 244h Odd Ports – SerDes Quad 2 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>Even/Odd Port SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <b>240h</b> through <b>24Ch</b>, in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port. 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]).</p> <p>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. 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.</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[8-11].</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.	RO	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.	RO	No	00h
23:16	<b>UTP/PRBS Error Counter</b> Receiver Detected flags. Returns the quantity of errors detected by the UTP (bit 30 is Cleared) or PRBS (bit 30 is Set) Data Checkers. The Error Counter saturates at 255, and is Cleared when UTP/PRBS is disabled. <b>UTP Mode</b> To Clear the Counter, disable UTP mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x User Test Pattern Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>258h</b> [23:16], and Odd Ports, offset <b>25Ch</b> [23:16]). <b>PRBS Mode</b> To Clear the Counter, disable PRBS mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x PRBS Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>258h</b> [31:24], and Odd Ports, offset <b>25Ch</b> [31:24]).	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-86. 244h Odd Ports – SerDes Quad 2 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<p><b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[8-11]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [15:0] 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] are selected with corresponding binary codes from 0-3.</p> <p><i>Note:</i> To obtain diagnostic data on all SerDes in the quad, run a test, then cycle these bits.</p>	RW	Yes	00b
29:26	<p><b>PRBS Error Injection</b> 1h = Causes the pattern generators for this SerDes quad's Lanes to insert an error into the transmitted pattern, once every 127 words</p>	RW	No	0h
30	<p><b>PRBS Counter-/UTP Counter</b> 0 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the UTP Error Counter 1 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the PRBS Error Counter</p>	RO	No	0
31	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-87. 248h Even Ports – SerDes Quad 1 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
There are four <b>Even/Odd Port SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <a href="#">240h</a> through <a href="#">24Ch</a> , in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port. 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]).				
	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. 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.			
	This register is used to retrieve Diagnostic Test results for SerDes[4-7].			
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.	RO	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.	RO	No	00h
23:16	<b>UTP/PRBS Error Counter</b> Receiver Detected flags. Returns the quantity of errors detected by the UTP (bit 30 is Cleared) or PRBS (bit 30 is Set) Data Checkers. The Error Counter saturates at 255, and is Cleared when UTP/PRBS is disabled. <b>UTP Mode</b> To Clear the Counter, disable UTP mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x User Test Pattern Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <a href="#">258h</a> [23:16], and Odd Ports, offset <a href="#">25Ch</a> [23:16]). <b>PRBS Mode</b> To Clear the Counter, disable PRBS mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x PRBS Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[31:24], and Odd Ports, offset 25Ch[31:24]).	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-87. 248h Even Ports – SerDes Quad 1 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<p><b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[4-7]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [15:0] 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] are selected with corresponding binary codes from 0-3.</p> <p><i>Note:</i> To obtain diagnostic data on all SerDes in the quad, run a test, then cycle these bits.</p>	RW	Yes	00b
29:26	<p><b>PRBS Error Injection</b> 1h = Causes the pattern generators for this SerDes quad's Lanes to insert an error into the transmitted pattern, once every 127 words</p>	RW	No	0h
30	<p><b>PRBS Counter-/UTP Counter</b> 0 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the UTP Error Counter 1 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the PRBS Error Counter</p>	RO	No	0
31	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-88. 24Ch Odd Ports – SerDes Quad 3 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>Even/Odd Port SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <b>240h</b> through <b>24Ch</b>, in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port. 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]).</p> <p>When field [25:24] is Cleared, the information in that Diagnostic Data register is for the first SerDes of that SerDes quad. When field [25:24] is programmed to 01b, the information in that Diagnostic Data register is for the second SerDes in each SerDes quad. Following this pattern, a value of 10b indicates the third SerDes in each SerDes quad, and a value of 11b indicates the fourth SerDes in each SerDes quad.</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[12-15].</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.	RO	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.	RO	No	00h
23:16	<b>UTP/PRBS Error Counter</b> Receiver Detected flags. Returns the quantity of errors detected by the UTP (bit 30 is Cleared) or PRBS (bit 30 is Set) Data Checkers. The Error Counter saturates at 255, and is Cleared when UTP/PRBS is disabled. <b>UTP Mode</b> To Clear the Counter, disable UTP mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x User Test Pattern Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>258h[23:16]</b> , and Odd Ports, offset <b>25Ch[23:16]</b> ). <b>PRBS Mode</b> To Clear the Counter, disable PRBS mode by Clearing one or more of the <b>Even/Odd Port User Test Pattern Delay</b> register <i>SerDes x PRBS Enable</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset <b>258h[31:24]</b> , and Odd Ports, offset <b>25Ch[31:24]</b> ).	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-88. 24Ch Odd Ports – SerDes Quad 3 Diagnostic Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<p><b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[12-15]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [15:0] 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] are selected with corresponding binary codes from 0-3.</p> <p><i>Note:</i> To obtain diagnostic data on all SerDes in the quad, run a test, then cycle these bits.</p>	RW	Yes	00b
29:26	<p><b>PRBS Error Injection</b> 1h = Causes the pattern generators for this SerDes quad's Lanes to insert an error into the transmitted pattern, once every 127 words</p>	RW	No	0h
30	<p><b>PRBS Counter-/UTP Counter</b> 0 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the UTP Error Counter 1 = Indicates that field [23:16] (<i>UTP/PRBS Error Counter</i>) is the PRBS Error Counter</p>	RO	No	0
31	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-89. 250h Even Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register contains PHY Safety bits that are associated with the Even Ports. It also contains Controls and Status for Link width re-configuration on Ports that are x8 or x4 capable in the Even PHY partition. Link width fields in this register are provided, to enable software to direct Link width Up/Down configuration. The Link width fields are loaded with the Target Link width, which should not exceed the originally negotiated Link width, and afterward, the <a href="#">Link Control</a> register <a href="#">Retrain Link</a> bit (offset 78h[5]) is Set. 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 re-negotiates the Link width.				
0	<p><b>Port 0 Downtain Disable</b> This bit supports Port 0, at up to x8 Link widths. 1 = Port 0 does not reduce its Link width during Link configuration, to eliminate faulty Lanes. Instead, the LTSSM returns to its <i>Detect</i> state and Retries Link training. This continues until the Link is able to train on all Lanes that detected a Receiver.</p> <p><i>Caution: This bit should be used with extreme caution, because its use could lead to an endless loop, in which the Link never reaches the L0 Link PM state.</i></p>	RWS	Yes	0
1	<p><b>Port 2 Downtain Disable</b> This bit supports Port 2, at up to x4 Link widths. 1 = Port 2 does not reduce its Link width during Link configuration, to eliminate faulty Lanes. Instead, the LTSSM returns to its <i>Detect</i> state and Retries Link training. This continues until the Link is able to train on all Lanes that detected a Receiver.</p> <p><i>Caution: This bit should be used with extreme caution, because its use could lead to an endless loop, in which the Link never reaches the L0 Link PM state.</i></p>	RWS	Yes	0
2	<p><b>Auto_Speed Disable</b> 1 = Upstream Port does not initiate a speed change</p>	RWS	Yes	0
3	<p><b>Enable x1 Even Port Skip Detect L0s Entry Mode</b> 1 = SKIP detection that is used for determining the end of L0s Link PM state Fast Training Sets on all Even Ports uses the Electrical Idle Inference SKIP Detector, instead of the standard detector located on the Deskew buffer output</p>	RWS	Yes	1
7:4	<p><b>Downstream Port 0 Hot Reset 2 ms Minimum Enable</b> 1h = If Port 0 is a downstream Port, it remains in the Hot Reset state for a minimum of 2 ms</p>	RWS	Yes	0h
11:8	<p><b>Downstream Port 2 Hot Reset 2 ms Minimum Enable</b> 1h = If Port 2 is a downstream Port, it remains in the Hot Reset state for a minimum of 2 ms</p>	RWS	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-89. 250h Even Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<b>Electrical Idle Inference Time Select[2:0]</b> For Even Ports, 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> state. 000b = 4 µs 001b = 6 µs 010b = 8 µs 011b = 16 µs 100b = 32 µs 101b = 64 µs 110b = 128 µs (default) 111b = 256 µs	RWS	Yes	110b
15	<b>Enable SSC Mode PhyStatus</b> 1 = For Even Ports, if SSC mode is enabled <b>STRAP_SSC_ISO_ENABLE#=L</b> , the PhyStatus of each Port 0 Lane is sourced only from Lane 0. This is true for any Port 0 Link width.	RWS	Yes	1
19:16	<b>Port 0 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 Port 0, to support Link Width Upconfiguration. The initial value of this field is the Port's <b>Negotiated Link Width</b> (offset <b>78h[25:20]</b> ).	RWS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>574h[3:0]</b> )

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-89. 250h Even Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Port 0 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 <b>Link Control</b> register <i>Retrain Link</i> bit (offset 78h[5]). 	RO	No	0
22:21	<b>Recovery.Speed Electrical Idle Inference Time Divider Select[1:0]</b> For Even Ports, selects the amount of time that no TS1 nor TS2 Ordered-Sets are detected during the <i>Recovery.Speed</i> state, for Electrical Idle to be inferred. (Refer to the <i>PCI Express Base r2.0</i> , Section 4.2.4.3, for the specific Unit Interval (UI) values.)  00b = <i>PCI Express Base r2.0</i> UI 01b = <i>PCI Express Base r2.0</i> UI/2 10b = <i>PCI Express Base r2.0</i> UI/4 11b = <i>PCI Express Base r2.0</i> UI/8	RWS	Yes	00b
23	<b>Even Ports Enable Elastic Buffer Reset on Recovery Entrance</b> 1 = Elastic buffer(s) associated with any Even Port are synchronously reset upon entering the LTSSM <i>Recovery</i> state	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-89. 250h Even Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 2 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 Port 2, to support Link Width Upconfiguration. The initial value of this field is the Port's <b>Negotiated Link Width</b> (offset 78h[25:20]).	RWS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
27, 31:29	<b>Configuration Fail Counter[3:0]</b> For Even Ports, specifies the quantity of times that the <i>Configuration</i> state must fail, before a Port toggles its Gen 2 Feature Disable flag. Writing 0000b to this field disables this Gen 1 compatibility function. The initial value of this field is determined by the <b>STRAP_UPCFG_TIMER_EN#</b> input state. If the input is Low when reset de-asserts, the initial value of this field is 0001b; otherwise, the initial value is 0000b.	RWS	Yes	0000b (STRAP_UPCFG_TIMER_EN#=H) 0001b (STRAP_UPCFG_TIMER_EN#=L)
28	<b>Port 2 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 <b>Link Control</b> register <b>Retrain Link</b> bit (offset 78h[5]).	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-90. 254h Odd Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register contains PHY Safety bits that are associated with the Odd Ports. It also contains Controls and Status for Link width re-configuration on Ports that are x8 or x4 capable in the Odd PHY partition.				
	Link width fields in this register are provided, to enable software to direct Link width Up/Down configuration. The Link width fields are loaded with the Target Link width, which should not exceed the originally negotiated Link width, and afterward, the <b>Link Control</b> register <i>Retrain Link</i> bit (offset 78h[5]) is Set. 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 re-negotiates the Link width.			
0	<p><b>Port 1 Downtain Disable</b> This bit supports Port 1, at up to x8 Link widths. 1 = Port 1 does not reduce its Link width during Link configuration, to eliminate faulty Lanes. Instead, the LTSSM returns to its <i>Detect</i> state and Retries Link training. This continues until the Link is able to train on all Lanes that detected a Receiver.</p> <p><i>Caution: This bit should be used with extreme caution, because its use could lead to an endless loop, in which the Link never reaches the L0 Link PM state.</i></p>	RWS	Yes	0
1	<p><b>Port 3 Downtain Disable</b> This bit supports Port 3, at up to x4 Link widths. 1 = Port 3 does not reduce its Link width during Link configuration, to eliminate faulty Lanes. Instead, the LTSSM returns to its <i>Detect</i> state and Retries Link training. This continues until the Link is able to train on all Lanes that detected a Receiver.</p> <p><i>Caution: This bit should be used with extreme caution, because its use could lead to an endless loop, in which the Link never reaches the L0 Link PM state.</i></p>	RWS	Yes	0
2	<p><b>Auto_Speed Disable</b> 1 = Upstream Port does not initiate a speed change</p>	RWS	Yes	0
3	<p><b>Enable x1 Odd Port Skip Detect L0s Entry Mode</b> 1 = SKIP detection that is used for determining the end of L0s Link PM state Fast Training Sets on all Odd Ports uses the Electrical Idle Inference SKIP Detector, instead of the standard detector located on the Deskew buffer output</p>	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-90. 254h Odd Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:4	<b>Downstream Port 1 Hot Reset 2 ms Minimum Enable</b> 1h = If Port 1 is a downstream Port, it remains in the Hot Reset state for a minimum of 2 ms	RWS	Yes	0h
11:8	<b>Downstream Port 3 Hot Reset 2 ms Minimum Enable</b> 1h = If Port 3 is a downstream Port, it remains in the Hot Reset state for a minimum of 2 ms	RWS	Yes	0h
14:12	<b>Electrical Idle Inference Time Select[2:0]</b> For Odd Ports, 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> state.  000b = 4 µs 001b = 6 µs 010b = 8 µs 011b = 16 µs 100b = 32 µs 101b = 64 µs 110b = 128 µs (default) 111b = 256 µs	RWS	Yes	110b
15	<b>Enable SSC Mode PhyStatus</b> This bit is <b>not</b> operational for Odd Ports. It is intended for use only by Even Port 0.	RWS	Yes	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-90. 254h Odd Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:16	<b>Port 1 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 Port 1, to support Link Width Upconfiguration. The initial value of this field is the Port's <b>Negotiated Link Width</b> (offset 78h[25:20]).	RWS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
20	<b>Port 1 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 <b>Link Control</b> register <i>Retrain Link</i> bit (offset 78h[5]).	RO	No	0
22:21	<b>Recovery Speed Electrical Idle Inference Time Divider Select[1:0]</b> For Odd Ports, selects the amount of time that no TS1 nor TS2 Ordered-Sets are detected during the <i>Recovery Speed</i> state, for Electrical Idle to be inferred. (Refer to the <i>PCI Express Base r2.0</i> , Section 4.2.4.3, for the specific Unit Interval (UI) values.)  00b = <i>PCI Express Base r2.0</i> UI 01b = <i>PCI Express Base r2.0</i> UI/2 10b = <i>PCI Express Base r2.0</i> UI/4 11b = <i>PCI Express Base r2.0</i> UI/8	RWS	Yes	00b
23	<b>Odd Ports Enable Elastic Buffer Reset on Recovery Entrance</b> 1 = Elastic buffer(s) associated with any Odd Port are synchronously reset upon entering the LTSSM <i>Recovery</i> state	RWS	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-90. 254h Odd Ports – Physical Layer Safety and Test  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 3 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 Port 3, to support Link Width Upconfiguration. The initial value of this field is the Port's <a href="#">Negotiated Link Width</a> (offset <a href="#">78h[25:20]</a> ).	RWS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <a href="#">Port Configuration</a> register <a href="#">Port Configuration</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
27, 31:29	<b>Configuration Fail Counter[3:0]</b> For Odd Ports, specifies the quantity of times that the <i>Configuration</i> state must fail, before a Port toggles its Gen 2 Feature Disable flag. Writing 0000b to this field disables this Gen 1 compatibility function. The initial value of this field is determined by the <a href="#">STRAP_UPCFG_TIMER_EN#</a> input state. If the input is Low when reset de-asserts, the initial value of this field is 0001b; otherwise, the initial value is 0000b.	RWS	Yes	0000b (STRAP_UPCFG_TIMER_EN#=H) 0001b (STRAP_UPCFG_TIMER_EN#=L)
28	<b>Port 3 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 (offset <a href="#">78h[5]</a> ).	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Lane 0 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 0. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
3:2	<b>Lane 1 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 1. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
5:4	<b>Lane 2 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 2. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
7:6	<b>Lane 3 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 3. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
9:8	<b>Lane 4 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 4. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
11:10	<b>Lane 5 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 5. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
13:12	<b>Lane 6 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 6. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
15:14	<b>Lane 7 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 7. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<p><b>SerDes 0 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 0 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 24 (SerDes 0 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 0. The logical result of bit 24 ANDed with bit 16 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
17	<p><b>SerDes 1 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 1 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 25 (SerDes 1 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 1. The logical result of bit 25 ANDed with bit 17 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18	<p><b>SerDes 2 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 2 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 26 (SerDes 2 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 2. The logical result of bit 26 ANDed with bit 18 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
19	<p><b>SerDes 3 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 3 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 27 (SerDes 3 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 3. The logical result of bit 27 ANDed with bit 19 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<p><b>SerDes 4 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 4 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 28 (SerDes 4 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 4. The logical result of bit 28 ANDed with bit 20 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
21	<p><b>SerDes 5 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 5 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 29 (SerDes 5 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 5. The logical result of bit 29 ANDed with bit 21 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
22	<p><b>SerDes 6 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 6 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 30 (SerDes 6 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 6. The logical result of bit 30 ANDed with bit 22 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
23	<p><b>SerDes 7 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 7 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 31 (SerDes 7 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 7. The logical result of bit 31 ANDed with bit 23 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>SerDes 0 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 0 1 = Enables PRBS sequence generation/checking on SerDes 0</p> <p><i>Notes: This bit and bit 16 (SerDes 0 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 0. The logical result of bit 16 ANDed with bit 24 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
25	<p><b>SerDes 1 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 1 1 = Enables PRBS sequence generation/checking on SerDes 1</p> <p><i>Notes: This bit and bit 17 (SerDes 1 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 1. The logical result of bit 17 ANDed with bit 25 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26	<p><b>SerDes 2 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 2 1 = Enables PRBS sequence generation/checking on SerDes 2</p> <p><i>Notes: This bit and bit 18 (SerDes 2 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 2. The logical result of bit 18 ANDed with bit 26 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
27	<p><b>SerDes 3 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 3 1 = Enables PRBS sequence generation/checking on SerDes 3</p> <p><i>Notes: This bit and bit 19 (SerDes 3 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 3. The logical result of bit 19 ANDed with bit 27 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>SerDes 4 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 4 1 = Enables PRBS sequence generation/checking on SerDes 4</p> <p><i>Notes: This bit and bit 20 (SerDes 4 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 4. The logical result of bit 20 ANDed with bit 28 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
29	<p><b>SerDes 5 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 5 1 = Enables PRBS sequence generation/checking on SerDes 5</p> <p><i>Notes: This bit and bit 21 (SerDes 5 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 5. The logical result of bit 21 ANDed with bit 29 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-91. 258h Even Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30	<p><b>SerDes 6 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 6 1 = Enables PRBS sequence generation/checking on SerDes 6</p> <p><i>Notes: This bit and bit 22 (SerDes 6 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 6. The logical result of bit 22 ANDed with bit 30 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
31	<p><b>SerDes 7 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 7 1 = Enables PRBS sequence generation/checking on SerDes 7</p> <p><i>Notes: This bit and bit 23 (SerDes 7 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 7. The logical result of bit 23 ANDed with bit 31 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Lane 8 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 8. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
3:2	<b>Lane 9 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 9. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
5:4	<b>Lane 10 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 10. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
7:6	<b>Lane 11 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 11. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
9:8	<b>Lane 12 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 12. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
11:10	<b>Lane 13 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 13. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
13:12	<b>Lane 14 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 14. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b
15:14	<b>Lane 15 User Test Pattern Delay</b> Selects the delay (in symbol times) applied when transmitting the UTP on Lane 15. 00b = 0 01b = 1 10b = 4 11b = 8	RW	Yes	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<p><b>SerDes 8 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 8 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 24 (SerDes 8 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 8. The logical result of bit 24 ANDed with bit 16 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
17	<p><b>SerDes 9 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 9 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 25 (SerDes 9 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 9. The logical result of bit 25 ANDed with bit 17 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18	<p><b>SerDes 10 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 10 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 26 (SerDes 10 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 10. The logical result of bit 26 ANDed with bit 18 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
19	<p><b>SerDes 11 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 11 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 27 (SerDes 11 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 11. The logical result of bit 27 ANDed with bit 19 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<p><b>SerDes 12 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 12 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 28 (SerDes 12 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 12. The logical result of bit 28 ANDed with bit 20 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
21	<p><b>SerDes 13 User Test Pattern Enable</b>            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 (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 13 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 29 (SerDes 13 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 13. The logical result of bit 29 ANDed with bit 21 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
22	<p><b>SerDes 14 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 14 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 30 (SerDes 14 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 14. The logical result of bit 30 ANDed with bit 22 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
23	<p><b>SerDes 15 User Test Pattern Enable</b>            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 registers</b> (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets <b>210h</b> through <b>21Ch</b>) on SerDes 15 in Digital Far-End Loopback Master mode</p> <p><i>Notes: This bit and bit 31 (SerDes 15 PRBS Enable) are mutually exclusive functions and must not be enabled together for SerDes 15. The logical result of bit 31 ANDed with bit 23 must be 0.</i></p> <p><i>UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>SerDes 8 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 8 1 = Enables PRBS sequence generation/checking on SerDes 8</p> <p><i>Notes: This bit and bit 16 (SerDes 8 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 8. The logical result of bit 16 ANDed with bit 24 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
25	<p><b>SerDes 9 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 9 1 = Enables PRBS sequence generation/checking on SerDes 9</p> <p><i>Notes: This bit and bit 17 (SerDes 9 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 9. The logical result of bit 17 ANDed with bit 25 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26	<p><b>SerDes 10 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 10 1 = Enables PRBS sequence generation/checking on SerDes 10</p> <p><i>Notes: This bit and bit 18 (SerDes 10 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 10. The logical result of bit 18 ANDed with bit 26 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
27	<p><b>SerDes 11 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 11 1 = Enables PRBS sequence generation/checking on SerDes 11</p> <p><i>Notes: This bit and bit 19 (SerDes 11 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 11. The logical result of bit 19 ANDed with bit 27 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>SerDes 12 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 12 1 = Enables PRBS sequence generation/checking on SerDes 12</p> <p><i>Notes: This bit and bit 20 (SerDes 12 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 12. The logical result of bit 20 ANDed with bit 28 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
29	<p><b>SerDes 13 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 13 1 = Enables PRBS sequence generation/checking on SerDes 13</p> <p><i>Notes: This bit and bit 21 (SerDes 13 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 13. The logical result of bit 21 ANDed with bit 29 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-92. 25Ch Odd Ports – User Test Pattern Delay  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30	<p><b>SerDes 14 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 14 1 = Enables PRBS sequence generation/checking on SerDes 14</p> <p><i>Notes: This bit and bit 22 (SerDes 14 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 14. The logical result of bit 22 ANDed with bit 30 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0
31	<p><b>SerDes 15 PRBS Enable</b> 0 = Disables PRBS sequence generation/checking on SerDes 15 1 = Enables PRBS sequence generation/checking on SerDes 15</p> <p><i>Notes: This bit and bit 23 (SerDes 15 User Test Pattern Enable) are mutually exclusive functions and must not be enabled together for SerDes 15. The logical result of bit 23 ANDed with bit 31 must be 0.</i></p> <p><i>PRBS transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet state and the Port's Even/Odd Port Quiet register Hold Port x Quiet bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8]) is Set.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.14.3 Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch)

This section details the Device-Specific Serial EEPROM registers. [Table 14-20](#) defines the register map.

**Table 14-20. Device-Specific Serial EEPROM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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																									
Serial EEPROM Buffer																																									
Serial EEPROM Clock Frequency																																									
Expansion ROM Base Address								<i>Reserved</i>								Serial EEPROM 3rd Address Byte																									

**Register 14-93. 260h Serial EEPROM Status and Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	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 = <b>Reserved</b> 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] ( <b>EepBlkAddr</b> ) 011b = Read four bytes of data from the memory location pointed to by field [12:0] ( <b>EepBlkAddr</b> ) 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 = <b>Reserved</b> <i>Note:</i> For value of 001b, only bits [31, 27:26] can be written into the serial EEPROM's internal <b>Status</b> register.	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-93. 260h Serial EEPROM Status and Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Serial EEPROM Status</b>				
17:16	<b>EepPrsnt[1:0]</b> Serial EEPROM Present status. 00b = Not present 01b = Serial EEPROM is present – validation signature verified 10b = <b>Reserved</b> 11b = Serial EEPROM is present – validation signature not verified	RO	No	00b
18	<b>EepCmdStatus</b> Serial EEPROM Command status. 0 = Serial EEPROM Command is complete 1 = Serial EEPROM Command is not complete	RO	No	0
19	<b>Reserved</b>	RsvdP	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 validation signature (5Ah) is successfully read from the first serial EEPROM location. This field is usually RO; 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	RO	No
		Bit 21 = 1	RW	No
				00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-93. 260h Serial EEPROM Status and Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																		
<b>Status Data from Serial EEPROM<sup>a</sup></b>																																						
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 8619 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. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th rowspan="2">BP[1:0]</th> <th rowspan="2">Level</th> <th colspan="4">Array Addresses Protected, by Device Size</th> </tr> <tr> <th>8 KB</th> <th>16 KB</th> <th>32 KB</th> <th>64-KB</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>0</td> <td>None</td> <td>None</td> <td>None</td> <td>None</td> </tr> <tr> <td>01b</td> <td>1 (top ¼)</td> <td>1800h – 1FFFh</td> <td>3000h – 3FFFh</td> <td>6000h – 7FFFh</td> <td>–</td> </tr> <tr> <td>10b</td> <td>2 (top ½)</td> <td>1000h – 1FFFh</td> <td>2000h – 3FFFh</td> <td>4000h – 7FFFh</td> <td>–</td> </tr> <tr> <td>11b</td> <td>3 (All)</td> <td>0000h – 1FFFh</td> <td>0000h – 3FFFh</td> <td>0000h – 7FFFh</td> <td>–</td> </tr> </tbody> </table>	BP[1:0]	Level	Array Addresses Protected, by Device Size				8 KB	16 KB	32 KB	64-KB	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	–	RW	Yes	00b
BP[1:0]	Level			Array Addresses Protected, by Device Size																																		
		8 KB	16 KB	32 KB	64-KB																																	
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	–																																	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-93. 260h Serial EEPROM Status and Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30:28	<p><b>EepWrStatus</b> Serial EEPROM Write status. Value is 000b when the serial EEPROM is not in an internal Write cycle.</p> <p><i>Note:</i> The definition of this field varies among serial EEPROM manufacturers. Reads of the serial EEPROM's internal <b>Status</b> register can return 000b or 111b, depending upon the serial EEPROM that is used.</p>	RO	No	000b
31	<p><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):</p> <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> <p><i>Notes:</i> 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.</p> <p>This bit is <b>not</b> implemented in certain serial EEPROMs. Refer to the serial EEPROM manufacturer's data sheet.</p>	RW	Yes	0

a. Within the serial EEPROM's internal **Status** register, only bits [31, 27:26] can be written.

**Register 14-94. 264h Serial EEPROM Buffer  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<p><b>EepBuf</b> Serial EEPROM RW buffer. Read/Write command to the <b>Serial EEPROM Control</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 260h) results in a 4-byte Read/Write to/from the serial EEPROM device.</p>	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-95. 268h Serial EEPROM Clock Frequency  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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 = <b>Reserved</b>	RW	Yes	000b
7:3	<b>Reserved</b>	RsvdP	No	0-0h
10:8	<b>EepCsStHld[2:0]</b> CS to SCLK setup and hold timing, provided as a quantity of $\frac{1}{2}$ EE_SK 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 $\frac{1}{2}$ EE_SK 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	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>Expansion ROM Size</b> 0 = 16 KB 1 = 32 KB	RW	Yes	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-96. 26Ch Serial EEPROM 3<sup>rd</sup> Address Byte  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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>	RW	Yes	00h
15:8	<b>Reserved</b>	RsvdP	No	00h
31:16	<p><b>Expansion ROM Base Address</b> Indicates the NT Port Expansion ROM Base address within the serial EEPROM. The value is dependent upon the <b>Serial EEPROM Clock Frequency</b> register <b>Expansion ROM Size</b> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 268h[16]) value.</p> <p>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.</p> <p>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.</p> <p>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.</p>	RW	Yes	0020h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

#### 14.14.4 Device-Specific Registers – Miscellaneous Control (Offset 28Ch)

This section details the Device-Specific Miscellaneous Control register. [Table 14-21](#) defines the register map.

**Table 14-21. Device-Specific Miscellaneous Control Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	<b>Miscellaneous Control</b>	28Ch
<i>Reserved</i>			

**Register 14-97. 28Ch Miscellaneous Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>VC1 (2nd VC) Disable</b> Use of VC1 is disabled, by default. To support its use, this bit must be Cleared. 0 = Enables VC1 to be used 1 = Disables VC1 from use	RWS	Yes	1
3:1	<b>Factory Test Only</b>	RWS	Yes	000b
7:4	<b>Reserved</b>	RsvdP	No	0h
15:8	<b>Factory Test Only</b>	RWS	Yes	00h
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 14.14.5 Device-Specific Registers – I<sup>2</sup>C Slave Interface (Offsets 290h – 2C4h)

This section details the Device-Specific I<sup>2</sup>C Slave Interface register. Table 14-22 defines the register map.

The I<sup>2</sup>C Slave Interface is described, in detail, in Section 7.2, “I<sup>2</sup>C Slave Interface.”

**Table 14-22. Device-Specific I<sup>2</sup>C Slave Interface Register Map (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
		290h
		294h
		298h – 2A8h
		2ACh
		2B0h
		2B4h – 2BCh
		2C0h – 2C4h

**Register 14-98. 294h I<sup>2</sup>C Configuration (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
2:0	<b>Slave Address</b> Bits [6:0] comprise the I <sup>2</sup> C/SMBus Slave address, 38h. The value is determined by bits [2:0] (which reflect the I <sub>2</sub> C_ADDR[2:0] input states, and default to 000b, by virtue of weak internal pull-up resistors), combined with the value of bits [6:3] (which default to 0111b). When I <sub>2</sub> C_ADDR2=H, Address Resolution Protocol (ARP) is disabled <b>and</b> bit 2 defaults to a value of 1. <i>Note: The I<sup>2</sup>C/SMBus Slave address must not be changed by an I<sup>2</sup>C/SMBus Write command.</i>	HwInit	Yes	000b	38h
6:3		RW	Yes	0111b	
9:7	<b>Reserved</b>	RsvdP	No	000b	
10	<b>Factory Test Only</b>	RW	Yes	0	
31:11	<b>Reserved</b>	RW	Yes	0-0h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.14.6 Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h)

This section details the Device-Specific Bus Number Content-Addressable Memory (BusNoCAM) registers, which are used to determine the Configuration TLP Completion routing. These registers contain mirror copies of the **Primary Bus Number**, **Secondary Bus Number**, and **Subordinate Bus Number** registers (offset 18h[23:16], 15:8, and 7:0), for each PEX 8619 Port. Table 14-23 defines the register map.

The BusNoCAM registers are automatically updated by hardware. *Modifying these registers by writing to the addresses listed here is not recommended.*

**Table 14-23. Device-Specific BusNoCAM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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>Reserved</i>		BusNoCAM0	2C8h
<i>Reserved</i>		BusNoCAM1	2CCh
<i>Reserved</i>		BusNoCAM2	2D0h
<i>Reserved</i>		BusNoCAM3	2D4h
<i>Reserved</i>		BusNoCAM4	2D8h
<i>Reserved</i>		BusNoCAM5	2DCh
<i>Reserved</i>		BusNoCAM6	2E0h
<i>Reserved</i>		BusNoCAM7	2E4h
<i>Reserved</i>		BusNoCAM8	2E8h
<i>Reserved</i>		BusNoCAM9	2ECh
<i>Reserved</i>		BusNoCAM10	2F0h
<i>Reserved</i>		BusNoCAM11	2F4h
<i>Reserved</i>		BusNoCAM12	2F8h
<i>Reserved</i>		BusNoCAM13	2FCCh
<i>Reserved</i>		BusNoCAM14	300h
<i>Reserved</i>		BusNoCAM15	304h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-99. 2C8h BusNoCAM0**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 0 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 0 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 0 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-100. 2CCh BusNoCAM1**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 1 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 1 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 1 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-101. 2D0h BusNoCAM2**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 2 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 2 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 2 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-102. 2D4h BusNoCAM3**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 3 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 3 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 3 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-103. 2D8h BusNoCAM4**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 4 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 4 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 4 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-104. 2DCh BusNoCAM5**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 5 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 5 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 5 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-105. 2E0h BusNoCAM6**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 6 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 6 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 6 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-106. 2E4h BusNoCAM7**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 7 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 7 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 7 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-107. 2E8h BusNoCAM8**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 8 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 8 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 8 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-108. 2ECh BusNoCAM9**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 9 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 9 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 9 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-109. 2F0h BusNoCAM10**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 10 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 10 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 10 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-110. 2F4h BusNoCAM11**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 11 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 11 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 11 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-111. 2F8h BusNoCAM12**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 12 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 12 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 12 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-112. 2FCh BusNoCAM13**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 13 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 13 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 13 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 14-113. 300h BusNoCAM14**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 14 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 14 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 14 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-114. 304h BusNoCAM15**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Mirror copy of Port 15 Primary Bus Number.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Mirror copy of Port 15 Secondary Bus Number.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Mirror copy of Port 15 Subordinate Bus Number.	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 14.14.7 Device-Specific Registers – I/O CAM (Offsets 308h – 340h)

This section details the Device-Specific I/O Content-Addressable Memory (IOCAM) registers, which are used to determine I/O Request routing. These registers contain mirror copies of the **I/O Base** and **I/O Limit** registers (offset 1Ch[7:0] and 15:8], respectively), for each PEX 8619 Port. Table 14-24 defines the register map.

The IOCAM registers are automatically updated by hardware. *Modifying these registers by writing to the addresses listed here is not recommended.*

**Table 14-24. Device-Specific IOCAM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
IOCAM1	IOCAM0	308h
IOCAM3	IOCAM2	30Ch
IOCAM5	IOCAM4	310h
IOCAM7	IOCAM6	314h
IOCAM9	IOCAM8	318h
IOCAM11	IOCAM10	31Ch
IOCAM13	IOCAM12	320h
IOCAM15	IOCAM14	324h
<i>Factory Test Only</i>		328h – 340h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-115. 308h IOCAM0**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 0 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 0 I/O Limit value.	RW	Yes	0h

**Register 14-116. 30Ah IOCAM1**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 1 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 1 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-117. 30Ch IOCAM2**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 2 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 2 I/O Limit value.	RW	Yes	0h

**Register 14-118. 30Eh IOCAM3**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 3 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 3 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-119. 310h IOCAM4**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 4 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 4 I/O Limit value.	RW	Yes	0h

**Register 14-120. 312h IOCAM5**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 5 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 5 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-121. 314h IOCAM6**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 6 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 6 I/O Limit value.	RW	Yes	0h

**Register 14-122. 316h IOCAM7**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 7 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 7 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-123. 318h IOCAM8**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 8 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 8 I/O Limit value.	RW	Yes	0h

**Register 14-124. 31Ah IOCAM9**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 9 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 9 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-125. 31Ch IOCAMP10**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 10 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 10 I/O Limit value.	RW	Yes	0h

**Register 14-126. 31Eh IOCAMP11**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 11 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 11 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-127. 320h IOCAM12**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 12 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 12 I/O Limit value.	RW	Yes	0h

**Register 14-128. 322h IOCAM13**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 13 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 13 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-129. 324h IOCAM14****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 14 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 14 I/O Limit value.	RW	Yes	0h

**Register 14-130. 326h IOCAM15****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
7:4	<b>I/O Base</b> Mirror copy of Port 15 I/O Base value.	RW	Yes	Fh
11:8	<b>I/O Addressing Capability</b> 0h = 16-bit I/O addressing 1h = 32-bit I/O addressing	RO	Yes	1h
15:12	<b>I/O Limit</b> Mirror copy of Port 15 I/O Limit value.	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.14.8 Device-Specific Registers – SMBus Slave Interface (Offset 344h)

This section details the Device-Specific SMBus Slave Interface register. Table 14-22 defines the register map.

The SMBus Slave Interface is described, in detail, in Section 7.3, “SMBus Slave Interface.”

**Table 14-25. Device-Specific SMBus Slave Interface Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	SMBus Configuration	344h
---	---------------------------------------	---------------------	------

**Register 14-131. 344h SMBus Configuration  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>SMBus Enable</b> Initially loaded from the STRAP_SMBUS_EN# input state. Value can later be changed by serial EEPROM and/or Configuration Space register Read Write. 0 = Disables SMBus for device configuration (I <sup>2</sup> C mode is enabled) 1 = Enables SMBus for device configuration (SMBus mode is enabled)	RW	Yes	0 (STRAP_SMBUS_EN#=H) 1 (STRAP_SMBUS_EN#=L)
7:1	<b>SMBus Device Address</b> Set by the Address Resolution Protocol (ARP), if ARP is enabled. If ARP is disabled (I2C_ADDR2=H), defaults to 38h, with Address bit [1:0] values loaded from the I2C_ADDR[1:0] input states.	RW	Yes	00h (I2C_ADDR2=L) 38h (I2C_ADDR2=H)
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	RW	Yes	0 (I2C_ADDR2=L) 1 (I2C_ADDR2=H)
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	RW	Yes	0
10	<b>AV Flag</b> Address Valid (AV) flag. Set, by default, when ARP is disabled (I2C_ADDR2=H).	RW	Yes	0 (I2C_ADDR2=L) 1 (I2C_ADDR2=H)
11	<b>AR Flag</b> Address Resolved (AR) flag.	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-131. 344h SMBus Configuration  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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 = I2C_ADDR2=H (ARP is disabled) 10b = I2C_ADDR2=L (ARP is enabled) All other encodings are <i>reserved</i> .	RW	Yes	00b (I2C_ADDR2=H) 10b (I2C_ADDR2=L)
14	<b>UDID PEC Support</b> 1 = Sets the <i>PEC Support</i> bit in the UDID	RW	Yes	1
15	<b>SMBus Parameter Reload</b> Set this bit if bits [10, 8, or 7:1] ( <i>AV Flag</i> , <i>ARP Disable</i> , or <i>SMBus Device Address</i> , respectively) are changed after a serial EEPROM load. Effective only when bit 28 ( <i>SMBus Command In-Progress</i> ) is Cleared.	RO	No	0
23:16	<b>UDID Vendor-Specific ID</b> Sets the MSB of the UDID Vendor-Specific ID. Bits [23:20] of this field are programmed by the I2C_ADDR[1:0] inputs. The four combinations provide the following ID values: 00b = 7000_0000h 01b = B000_0000h 10b = D000_0000h 11b = E000_0000h	RW	Yes	Set by I2C_ADDR[1:0] input levels
26:24	<b>UDID Revision ID</b> Program to 001b for Silicon Revision BA.	RW	Yes	001b
27	<b>Factory Test Only</b>	RW	Yes	0
28	<b>SMBus Command In-Progress</b> 0 = SMBus state machine is idle 1 = SMBus state machine is active (not idle)	RO	No	0
29	<b>PEC Check Failed</b> 0 = PEC checking successfully completed when receiving a packet 1 = PEC checking 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>Factory Test Only</b>	RW1C	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.14.9 Device-Specific Registers – Address-Mapping CAM (Offsets 348h – 444h)

This section details the Device-Specific Address-Mapping Content-Addressable Memory (AMCAM) registers, which are used to determine Memory Request routing. These registers contain mirror copies of the **Memory Base and Limit**, **Prefetchable Memory Base and Limit**, **Prefetchable Memory Upper Base Address**, and **Prefetchable Memory Upper Limit Address** registers (offsets 20h, 24h, 28h, and 2Ch, respectively), for each PEX 8619 Port. Table 14-26 defines the register map.

The AMCAM registers are automatically updated by hardware. *Modifying these registers by writing to the addresses listed here is not recommended.*

**Table 14-26. Device-Specific AMCAM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
AMCAM0 Memory Limit		AMCAM0 Memory Base	348h
AMCAM0 Prefetchable Memory Limit		AMCAM0 Prefetchable Memory Base	34Ch
	AMCAM0 Prefetchable Memory Base Upper 32 Bits		350h
	AMCAM0 Prefetchable Memory Limit Upper 32 Bits		354h
AMCAM1 Memory Limit		AMCAM1 Memory Base	358h
AMCAM1 Prefetchable Memory Limit		AMCAM1 Prefetchable Memory Base	35Ch
	AMCAM1 Prefetchable Memory Base Upper 32 Bits		360h
	AMCAM1 Prefetchable Memory Limit Upper 32 Bits		364h
AMCAM2 Memory Limit		AMCAM2 Memory Base	368h
AMCAM2 Prefetchable Memory Limit		AMCAM2 Prefetchable Memory Base	36Ch
	AMCAM2 Prefetchable Memory Base Upper 32 Bits		370h
	AMCAM2 Prefetchable Memory Limit Upper 32 Bits		374h
AMCAM3 Memory Limit		AMCAM3 Memory Base	378h
AMCAM3 Prefetchable Memory Limit		AMCAM3 Prefetchable Memory Base	37Ch
	AMCAM3 Prefetchable Memory Base Upper 32 Bits		380h
	AMCAM3 Prefetchable Memory Limit Upper 32 Bits		384h
AMCAM4 Memory Limit		AMCAM4 Memory Base	388h
AMCAM4 Prefetchable Memory Limit		AMCAM4 Prefetchable Memory Base	38Ch
	AMCAM4 Prefetchable Memory Base Upper 32 Bits		390h
	AMCAM4 Prefetchable Memory Limit Upper 32 Bits		394h
AMCAM5 Memory Limit		AMCAM5 Memory Base	398h
AMCAM5 Prefetchable Memory Limit		AMCAM5 Prefetchable Memory Base	39Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 14-26. Device-Specific AMCAM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
	AMCAM5 Prefetchable Memory Base Upper 32 Bits	3A0h
	AMCAM5 Prefetchable Memory Limit Upper 32 Bits	3A4h
AMCAM6 Memory Limit	AMCAM6 Memory Base	3A8h
AMCAM6 Prefetchable Memory Limit	AMCAM6 Prefetchable Memory Base	3ACh
	AMCAM6 Prefetchable Memory Base Upper 32 Bits	3B0h
	AMCAM6 Prefetchable Memory Limit Upper 32 Bits	3B4h
AMCAM7 Memory Limit	AMCAM7 Memory Base	3B8h
AMCAM7 Prefetchable Memory Limit	AMCAM7 Prefetchable Memory Base	3BCh
	AMCAM7 Prefetchable Memory Base Upper 32 Bits	3C0h
	AMCAM7 Prefetchable Memory Limit Upper 32 Bits	3C4h
AMCAM8 Memory Limit	AMCAM8 Memory Base	3C8h
AMCAM8 Prefetchable Memory Limit	AMCAM8 Prefetchable Memory Base	3CCh
	AMCAM8 Prefetchable Memory Base Upper 32 Bits	3D0h
	AMCAM8 Prefetchable Memory Limit Upper 32 Bits	3D4h
AMCAM9 Memory Limit	AMCAM9 Memory Base	3D8h
AMCAM9 Prefetchable Memory Limit	AMCAM9 Prefetchable Memory Base	3DCh
	AMCAM9 Prefetchable Memory Base Upper 32 Bits	3E0h
	AMCAM9 Prefetchable Memory Limit Upper 32 Bits	3E4h
AMCAM10 Memory Limit	AMCAM10 Memory Base	3E8h
AMCAM10 Prefetchable Memory Limit	AMCAM10 Prefetchable Memory Base	3ECh
	AMCAM10 Prefetchable Memory Base Upper 32 Bits	3F0h
	AMCAM10 Prefetchable Memory Limit Upper 32 Bits	3F4h
AMCAM11 Memory Limit	AMCAM11 Memory Base	3F8h
AMCAM11 Prefetchable Memory Limit	AMCAM11 Prefetchable Memory Base	3FCh
	AMCAM11 Prefetchable Memory Base Upper 32 Bits	400h
	AMCAM11 Prefetchable Memory Limit Upper 32 Bits	404h
AMCAM12 Memory Limit	AMCAM12 Memory Base	408h
AMCAM12 Prefetchable Memory Limit	AMCAM12 Prefetchable Memory Base	40Ch
	AMCAM12 Prefetchable Memory Base Upper 32 Bits	410h
	AMCAM12 Prefetchable Memory Limit Upper 32 Bits	414h
AMCAM13 Memory Limit	AMCAM13 Memory Base	418h
AMCAM13 Prefetchable Memory Limit	AMCAM13 Prefetchable Memory Base	41Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-26. Device-Specific AMCAM Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
	AMCAM13 Prefetchable Memory Base Upper 32 Bits		420h
	AMCAM13 Prefetchable Memory Limit Upper 32 Bits		424h
AMCAM14 Memory Limit		AMCAM14 Memory Base	428h
AMCAM14 Prefetchable Memory Limit		AMCAM14 Prefetchable Memory Base	42Ch
	AMCAM14 Prefetchable Memory Base Upper 32 Bits		430h
	AMCAM14 Prefetchable Memory Limit Upper 32 Bits		434h
AMCAM15 Memory Limit		AMCAM15 Memory Base	438h
AMCAM15 Prefetchable Memory Limit		AMCAM15 Prefetchable Memory Base	43Ch
	AMCAM15 Prefetchable Memory Base Upper 32 Bits		440h
	AMCAM15 Prefetchable Memory Limit Upper 32 Bits		444h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-132. 348h AMCAM0 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM0 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM0 Memory Base</b> Mirror copy of Port 0 Memory Base value.	RW	Yes	FFFh
<b>AMCAM0 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM0 Memory Limit</b> Mirror copy of Port 0 Memory Limit value.	RW	Yes	000h

**Register 14-133. 34Ch AMCAM0 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM0 Prefetchable Memory Base</b>				
3:0	<b>AMCAM0 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM0 Prefetchable Memory Base</b> AMCAM0 Port 0 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM0 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM0 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM0 Prefetchable Memory Limit</b> AMCAM0 Port 0 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-134. 350h AMCAM0 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM0 Prefetchable Memory Base Upper 32 Bits</b> AMCAM0 Port 0 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-135. 354h AMCAM0 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM0 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM0 Port 0 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-136. 358h AMCAM1 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM1 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM1 Memory Base</b> Mirror copy of Port 1 Memory Base value.	RW	Yes	FFFh
<b>AMCAM1 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM1 Memory Limit</b> Mirror copy of Port 1 Memory Limit value.	RW	Yes	000h

**Register 14-137. 35Ch AMCAM1 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM1 Prefetchable Memory Base</b>				
3:0	<b>AMCAM1 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM1 Prefetchable Memory Base</b> AMCAM1 Port 1 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM1 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM1 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM1 Prefetchable Memory Limit</b> AMCAM1 Port 1 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-138. 360h AMCAM1 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM1 Prefetchable Memory Base Upper 32 Bits</b> AMCAM1 Port 1 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-139. 364h AMCAM1 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM1 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM1 Port 1 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-140. 368h AMCAM2 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM2 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM2 Memory Base</b> Mirror copy of Port 2 Memory Base value.	RW	Yes	FFFh
<b>AMCAM2 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM2 Memory Limit</b> Mirror copy of Port 2 Memory Limit value.	RW	Yes	000h

**Register 14-141. 36Ch AMCAM2 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM2 Prefetchable Memory Base</b>				
3:0	<b>AMCAM2 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM2 Prefetchable Memory Base</b> AMCAM2 Port 2 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM2 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM2 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM2 Prefetchable Memory Limit</b> AMCAM2 Port 2 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-142. 370h AMCAM2 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM2 Prefetchable Memory Base Upper 32 Bits</b> AMCAM2 Port 2 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-143. 374h AMCAM2 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM2 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM2 Port 2 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-144. 378h AMCAM3 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM3 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM3 Memory Base</b> Mirror copy of Port 3 Memory Base value.	RW	Yes	FFFh
<b>AMCAM3 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM3 Memory Limit</b> Mirror copy of Port 3 Memory Limit value.	RW	Yes	000h

**Register 14-145. 37Ch AMCAM3 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM3 Prefetchable Memory Base</b>				
3:0	<b>AMCAM3 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM3 Prefetchable Memory Base</b> AMCAM3 Port 3 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM3 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM3 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM3 Prefetchable Memory Limit</b> AMCAM3 Port 3 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-146. 380h AMCAM3 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM3 Prefetchable Memory Base Upper 32 Bits</b> AMCAM3 Port 3 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-147. 384h AMCAM3 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM3 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM3 Port 3 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-148. 388h AMCAM4 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM4 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM4 Memory Base</b> Mirror copy of Port 4 Memory Base value.	RW	Yes	FFFh
<b>AMCAM4 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM4 Memory Limit</b> Mirror copy of Port 4 Memory Limit value.	RW	Yes	000h

**Register 14-149. 38Ch AMCAM4 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM4 Prefetchable Memory Base</b>				
3:0	<b>AMCAM4 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM4 Prefetchable Memory Base</b> AMCAM4 Port 4 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM4 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM4 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM4 Prefetchable Memory Limit</b> AMCAM4 Port 4 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-150. 390h AMCAM4 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM4 Prefetchable Memory Base Upper 32 Bits</b> AMCAM4 Port 4 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-151. 394h AMCAM4 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM4 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM4 Port 4 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-152. 398h AMCAM5 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM5 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM5 Memory Base</b> Mirror copy of Port 5 Memory Base value.	RW	Yes	FFFh
<b>AMCAM5 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM5 Memory Limit</b> Mirror copy of Port 5 Memory Limit value.	RW	Yes	000h

**Register 14-153. 39Ch AMCAM5 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM5 Prefetchable Memory Base</b>				
3:0	<b>AMCAM5 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM5 Prefetchable Memory Base</b> AMCAM5 Port 5 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM5 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM5 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM5 Prefetchable Memory Limit</b> AMCAM5 Port 5 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-154. 3A0h AMCAM5 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM5 Prefetchable Memory Base Upper 32 Bits</b> AMCAM5 Port 5 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-155. 3A4h AMCAM5 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM5 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM5 Port 5 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-156. 3A8h AMCAM6 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM6 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM6 Memory Base</b> Mirror copy of Port 6 Memory Base value.	RW	Yes	FFFh
<b>AMCAM6 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM6 Memory Limit</b> Mirror copy of Port 6 Memory Limit value.	RW	Yes	000h

**Register 14-157. 3ACh AMCAM6 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM6 Prefetchable Memory Base</b>				
3:0	<b>AMCAM6 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM6 Prefetchable Memory Base</b> AMCAM6 Port 6 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM6 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM6 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM6 Prefetchable Memory Limit</b> AMCAM6 Port 6 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-158. 3B0h AMCAM6 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM6 Prefetchable Memory Base Upper 32 Bits</b> AMCAM6 Port 6 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-159. 3B4h AMCAM6 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM6 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM6 Port 6 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-160. 3B8h AMCAM7 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM7 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM7 Memory Base</b> Mirror copy of Port 7 Memory Base value.	RW	Yes	FFFh
<b>AMCAM7 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM7 Memory Limit</b> Mirror copy of Port 7 Memory Limit value.	RW	Yes	000h

**Register 14-161. 3BCh AMCAM7 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM7 Prefetchable Memory Base</b>				
3:0	<b>AMCAM7 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM7 Prefetchable Memory Base</b> AMCAM7 Port 7 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM7 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM7 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM7 Prefetchable Memory Limit</b> AMCAM7 Port 7 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-162. 3C0h AMCAM7 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM7 Prefetchable Memory Base Upper 32 Bits</b> AMCAM7 Port 7 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-163. 3C4h AMCAM7 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM7 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM7 Port 7 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-164. 3C8h AMCAM8 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM8 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM8 Memory Base</b> Mirror copy of Port 8 Memory Base value.	RW	Yes	FFFh
<b>AMCAM8 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM8 Memory Limit</b> Mirror copy of Port 8 Memory Limit value.	RW	Yes	000h

**Register 14-165. 3CCh AMCAM8 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM8 Prefetchable Memory Base</b>				
3:0	<b>AMCAM8 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM8 Prefetchable Memory Base</b> AMCAM8 Port 8 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM8 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM8 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM8 Prefetchable Memory Limit</b> AMCAM8 Port 8 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-166. 3D0h AMCAM8 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM8 Prefetchable Memory Base Upper 32 Bits</b> AMCAM8 Port 8 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-167. 3D4h AMCAM8 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM8 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM8 Port 8 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-168. 3D8h AMCAM9 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM9 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM9 Memory Base</b> Mirror copy of Port 9 Memory Base value.	RW	Yes	FFFh
<b>AMCAM9 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM9 Memory Limit</b> Mirror copy of Port 9 Memory Limit value.	RW	Yes	000h

**Register 14-169. 3DCh AMCAM9 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM9 Prefetchable Memory Base</b>				
3:0	<b>AMCAM9 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM9 Prefetchable Memory Base</b> AMCAM9 Port 9 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM9 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM9 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM9 Prefetchable Memory Limit</b> AMCAM9 Port 9 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-170. 3E0h AMCAM9 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM9 Prefetchable Memory Base Upper 32 Bits</b> AMCAM9 Port 9 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-171. 3E4h AMCAM9 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM9 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM9 Port 9 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-172. 3E8h AMCAM10 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM10 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM10 Memory Base</b> Mirror copy of Port 10 Memory Base value.	RW	Yes	FFFh
<b>AMCAM10 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM10 Memory Limit</b> Mirror copy of Port 10 Memory Limit value.	RW	Yes	000h

**Register 14-173. 3EC<sub>h</sub> AMCAM10 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM10 Prefetchable Memory Base</b>				
3:0	<b>AMCAM10 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM10 Prefetchable Memory Base</b> AMCAM10 Port 10 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM10 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM10 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM10 Prefetchable Memory Limit</b> AMCAM10 Port 10 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-174. 3F0<sub>h</sub> AMCAM10 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM10 Prefetchable Memory Base Upper 32 Bits</b> AMCAM10 Port 10 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-175. 3F4<sub>h</sub> AMCAM10 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM10 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM10 Port 10 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-176. 3F8h AMCAM11 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM11 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM11 Memory Base</b> Mirror copy of Port 11 Memory Base value.	RW	Yes	FFFh
<b>AMCAM11 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM11 Memory Limit</b> Mirror copy of Port 11 Memory Limit value.	RW	Yes	000h

**Register 14-177. 3FCh AMCAM11 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM11 Prefetchable Memory Base</b>				
3:0	<b>AMCAM11 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM11 Prefetchable Memory Base</b> AMCAM11 Port 11 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM11 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM11 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM11 Prefetchable Memory Limit</b> AMCAM11 Port 11 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-178. 400h AMCAM11 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM11 Prefetchable Memory Base Upper 32 Bits</b> AMCAM11 Port 11 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-179. 404h AMCAM11 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM11 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM11 Port 11 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-180. 408h AMCAM12 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM12 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM12 Memory Base</b> Mirror copy of Port 12 Memory Base value.	RW	Yes	FFFh
<b>AMCAM12 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM12 Memory Limit</b> Mirror copy of Port 12 Memory Limit value.	RW	Yes	000h

**Register 14-181. 40Ch AMCAM12 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM12 Prefetchable Memory Base</b>				
3:0	<b>AMCAM12 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM12 Prefetchable Memory Base</b> AMCAM12 Port 12 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM12 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM12 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM12 Prefetchable Memory Limit</b> AMCAM12 Port 12 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-182. 410h AMCAM12 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM12 Prefetchable Memory Base Upper 32 Bits</b> AMCAM12 Port 12 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-183. 414h AMCAM12 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM12 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM12 Port 12 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-184. 418h AMCAM13 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM13 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM13 Memory Base</b> Mirror copy of Port 13 Memory Base value.	RW	Yes	FFFh
<b>AMCAM13 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM13 Memory Limit</b> Mirror copy of Port 13 Memory Limit value.	RW	Yes	000h

**Register 14-185. 41Ch AMCAM13 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM13 Prefetchable Memory Base</b>				
3:0	<b>AMCAM13 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM13 Prefetchable Memory Base</b> AMCAM13 Port 13 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM13 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM13 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM13 Prefetchable Memory Limit</b> AMCAM13 Port 13 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-186. 420h AMCAM13 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM13 Prefetchable Memory Base Upper 32 Bits</b> AMCAM13 Port 13 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-187. 424h AMCAM13 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM13 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM13 Port 13 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-188. 428h AMCAM14 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM14 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM14 Memory Base</b> Mirror copy of Port 14 Memory Base value.	RW	Yes	FFFh
<b>AMCAM14 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM14 Memory Limit</b> Mirror copy of Port 14 Memory Limit value.	RW	Yes	000h

**Register 14-189. 42Ch AMCAM14 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM14 Prefetchable Memory Base</b>				
3:0	<b>AMCAM14 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM14 Prefetchable Memory Base</b> AMCAM14 Port 14 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM14 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM14 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM14 Prefetchable Memory Limit</b> AMCAM14 Port 14 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-190. 430h AMCAM14 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM14 Prefetchable Memory Base Upper 32 Bits</b> AMCAM14 Port 14 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-191. 434h AMCAM14 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM14 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM14 Port 14 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-192. 438h AMCAM15 Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM15 Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>AMCAM15 Memory Base</b> Mirror copy of Port 15 Memory Base value.	RW	Yes	FFFh
<b>AMCAM15 Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>AMCAM15 Memory Limit</b> Mirror copy of Port 15 Memory Limit value.	RW	Yes	000h

**Register 14-193. 43Ch AMCAM15 Prefetchable Memory Base and Limit  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>AMCAM15 Prefetchable Memory Base</b>				
3:0	<b>AMCAM15 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
15:4	<b>AMCAM15 Prefetchable Memory Base</b> AMCAM15 Port 15 Prefetchable Memory Base[31:20].	RW	Yes	FFFh
<b>AMCAM15 Prefetchable Memory Limit</b>				
19:16	<b>AMCAM15 Addressing Support</b> 0h = 32-bit addressing is supported 1h = 64-bit addressing is supported	RO	Yes	1h
31:20	<b>AMCAM15 Prefetchable Memory Limit</b> AMCAM15 Port 15 Prefetchable Memory Limit[31:20].	RW	Yes	000h

**Register 14-194. 440h AMCAM15 Prefetchable Memory Base Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM15 Prefetchable Memory Base Upper 32 Bits</b> AMCAM15 Port 15 Prefetchable Memory Base[63:32].	RW	Yes	0000_0000h

**Register 14-195. 444h AMCAM15 Prefetchable Memory Limit Upper 32 Bits  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>AMCAM15 Prefetchable Memory Limit Upper 32 Bits</b> AMCAM15 Port 15 Prefetchable Memory Limit[63:32].	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 14.14.10 Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability (Offsets 448h – 51Ch)

This section details the Device-Specific, Vendor-Specific Dual Cast Extended Capability registers. [Table 14-27](#) defines the register map.

**Table 14-27. Device-Specific, Vendor-Specific Dual Cast Extended Capability Register Map (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
Next Capability Offset (950h or 520h)	Capability Version (1h)	PCI Express Extended Capability ID (000Bh)
		448h
	Vendor-Specific Header 1	44Ch
	Dual Cast Low BAR0	450h
	Dual Cast High BAR0	454h
	Dual Cast Low BAR0 Translation	458h
	Dual Cast High BAR0 Translation	45Ch
	Dual Cast Low BAR0 Setup	460h
	Dual Cast High BAR0 Setup	464h
	Dual Cast Low BAR1	468h
	Dual Cast High BAR1	46Ch
	Dual Cast Low BAR1 Translation	470h
	Dual Cast High BAR1 Translation	474h
	Dual Cast Low BAR1 Setup	478h
	Dual Cast High BAR1 Setup	47Ch
	Dual Cast Low BAR2	480h
	Dual Cast High BAR2	484h
	Dual Cast Low BAR2 Translation	488h
	Dual Cast High BAR2 Translation	48Ch
	Dual Cast Low BAR2 Setup	490h
	Dual Cast High BAR2 Setup	494h
	Dual Cast Low BAR3	498h
	Dual Cast High BAR3	49Ch
	Dual Cast Low BAR3 Translation	4A0h
	Dual Cast High BAR3 Translation	4A4h
	Dual Cast Low BAR3 Setup	4A8h
	Dual Cast High BAR3 Setup	4ACh
	Dual Cast Low BAR4	4B0h
	Dual Cast High BAR4	4B4h
	Dual Cast Low BAR4 Translation	4B8h
	Dual Cast High BAR4 Translation	4BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-27. Device-Specific, Vendor-Specific Dual Cast Extended Capability Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
	Dual Cast Low BAR4 Setup	4C0h
	Dual Cast High BAR4 Setup	4C4h
	Dual Cast Low BAR5	4C8h
	Dual Cast High BAR5	4CCh
	Dual Cast Low BAR5 Translation	4D0h
	Dual Cast High BAR5 Translation	4D4h
	Dual Cast Low BAR5 Setup	4D8h
	Dual Cast High BAR5 Setup	4DCh
	Dual Cast Low BAR6	4E0h
	Dual Cast High BAR6	4E4h
	Dual Cast Low BAR6 Translation	4E8h
	Dual Cast High BAR6 Translation	4ECh
	Dual Cast Low BAR6 Setup	4F0h
	Dual Cast High BAR6 Setup	4F4h
	Dual Cast Low BAR7	4F8h
	Dual Cast High BAR7	4FCh
	Dual Cast Low BAR7 Translation	500h
	Dual Cast High BAR7 Translation	504h
	Dual Cast Low BAR7 Setup	508h
	Dual Cast High BAR7 Setup	50Ch
	Dual Cast Source Destination Port	510h
	<i>Reserved</i>	514h – 51Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-196. 448h Vendor-Specific Extended Capability Header  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

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 000Bh, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.		RO	Yes	000Bh
19:16	<b>Capability Version</b>		RO	Yes	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>Vendor-Specific Extended Capability 2</b> structure, offset 950h.	Upstream	RO	Yes	950h
	Next extended capability is the <b>ACS Extended Capability</b> structure, offset 520h.	Downstream	RO	No	520h

**Register 14-197. 44Ch Vendor-Specific Header 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor-Specific ID</b> ID Number of this Extended Capability structure.	RO	Yes	0000h
19:16	<b>Vendor-Specific Rev</b> Version Number of this structure.	RO	Yes	0h
31:20	<b>Vendor-Specific Length</b> Quantity of bytes in the entire structure.	RO	Yes	0CCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-198. 450h Dual Cast Low BAR0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Prefetchable</b> Not used, but included for software compatibility with respect to Base Address Register (BAR) definition.	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-199. 454h Dual Cast High BAR0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0-0h

**Register 14-200. 458h Dual Cast Low BAR0 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-201. 45Ch Dual Cast High BAR0 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR0</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 450h[2:1]) is programmed to 10b.	RW	Yes	0-0h
	<b>Reserved</b> when the <b>Dual Cast Low BAR0</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 450h[2:1]) is Cleared.	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-202. 460h Dual Cast Low BAR0 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-203. 464h Dual Cast High BAR0 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR0</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 450h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0-0h
	<b>Reserved</b> when the <b>Dual Cast Low BAR0</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 450h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-204. 468h Dual Cast Low BAR1**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-205. 46Ch Dual Cast High BAR1**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-206. 470h Dual Cast Low BAR1 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-207. 474h Dual Cast High BAR1 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR1</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 468h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR1</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 468h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-208. 478h Dual Cast Low BAR1 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-209. 47Ch Dual Cast High BAR1 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR1</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 468h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR1</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 468h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-210. 480h Dual Cast Low BAR2**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-211. 484h Dual Cast High BAR2**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-212. 488h Dual Cast Low BAR2 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-213. 48Ch Dual Cast High BAR2 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR2</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 480h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR2</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 480h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-214. 490h Dual Cast Low BAR2 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-215. 494h Dual Cast High BAR2 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR2</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 480h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR2</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 480h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-216. 498h Dual Cast Low BAR3**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-217. 49Ch Dual Cast High BAR3**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-218. 4A0h Dual Cast Low BAR3 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-219. 4A4h Dual Cast High BAR3 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR3</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 498h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR3</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 498h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-220. 4A8h Dual Cast Low BAR3 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-221. 4ACh Dual Cast High BAR3 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR3</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 498h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR3</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 498h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-222. 4B0h Dual Cast Low BAR4**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-223. 4B4h Dual Cast High BAR4**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-224. 4B8h Dual Cast Low BAR4 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-225. 4BCCh Dual Cast High BAR4 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR4</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4B0h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR4</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4B0h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-226. 4C0h Dual Cast Low BAR4 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-227. 4C4h Dual Cast High BAR4 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR4</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4B0h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR4</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4B0h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-228. 4C8h Dual Cast Low BAR5**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-229. 4CCh Dual Cast High BAR5**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-230. 4D0h Dual Cast Low BAR5 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-231. 4D4h Dual Cast High BAR5 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR5</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4C8h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR5</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4C8h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-232. 4D8h Dual Cast Low BAR5 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-233. 4DCh Dual Cast High BAR5 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR5</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4C8h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR5</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4C8h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-234. 4E0h Dual Cast Low BAR6**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-235. 4E4h Dual Cast High BAR6**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-236. 4E8h Dual Cast Low BAR6 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-237. 4ECh Dual Cast High BAR6 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR6</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4E0h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR6</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4E0h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-238. 4F0h Dual Cast Low BAR6 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-239. 4F4h Dual Cast High BAR6 Setup  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR6</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4E0h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR6</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4E0h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-240. 4F8h Dual Cast Low BAR7**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	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> .	RO	No	10b
3	<b>Reserved</b>	RO	Yes	1
19:4	<b>Reserved</b>	RO	No	0_000h
31:20	<b>Dual Cast BAR</b>	RW	Yes	000h

**Register 14-241. 4FCh Dual Cast High BAR7**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper Base Address</b> Dual Cast's Upper Base Address (BAR[63:32]).	RW	No	0000_0000h

**Register 14-242. 500h Dual Cast Low BAR7 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Translation Address</b>	RW	Yes	000h

**Register 14-243. 504h Dual Cast High BAR7 Translation**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Base Translation Upper Address</b> RW when the <b>Dual Cast Low BAR7</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4F8h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR7</b> register <b>Memory Map Type</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4F8h[2:1]) is Cleared.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-244. 508h Dual Cast Low BAR7 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	RsvdP	No	0_0000h
31:20	<b>Base Setup</b>	RW	Yes	000h

**Register 14-245. 50Ch Dual Cast High BAR7 Setup**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Dual Cast Upper BAR Setup</b> RW when the <b>Dual Cast Low BAR7</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4F8h[2:1]) is programmed to 10b. Bit 31 values: 0 = BAR is disabled 1 = BAR is enabled	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>Dual Cast Low BAR7</b> register <i>Memory Map Type</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 4F8h[2:1]) is Cleared.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-246. 510h Dual Cast Source Destination Port  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port;  
Reserved (RsvdP) for the Dual Cast BAR Limit Lower register)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Dual Cast Source Port #</b>            Valid only when bit 8 (<i>Dual Cast Source Port Enable</i>) is Set.            Specifies the source (ingress) Port Number upon which Dual Cast BARs are applied.</p> <p>0h = Port 0            1h = Port 1            2h = Port 2            3h = Port 3            4h = Port 4            5h = Port 5            6h = Port 6            7h = Port 7            8h = Port 8            9h = Port 9            Ah = Port 10            Bh = Port 11            Ch = Port 12            Dh = Port 13            Eh = Port 14            Fh = Port 15</p>	RW	Yes	0h
7:4	<p><b>Dual Cast Destination Port #</b>            Specifies the destination (egress) Port Number to which Dual Cast BAR Translation addresses are mapped, and to which Dual Cast Copy TLPs will be queued.</p> <p>0h = Port 0            1h = Port 1            2h = Port 2            3h = Port 3            4h = Port 4            5h = Port 5            6h = Port 6            7h = Port 7            8h = Port 8            9h = Port 9            Ah = Port 10            Bh = Port 11            Ch = Port 12            Dh = Port 13            Eh = Port 14            Fh = Port 15</p>	RW	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-246. 510h Dual Cast Source Destination Port  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port;  
Reserved (RsvdP) for the Dual Cast BAR Limit Lower register) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Dual Cast Source Port Enable</b> 0 = Dual Cast applies to Write TLPs entering any Port on the Dual Cast Source Port. In that case, the field [3:0] ( <i>Dual Cast Source Port #</i> ) value is “Don’t Care.” 1 = Dual Cast applies only to Write TLPs entering the PEX 8619, by way of the Dual Cast Source Port Number specified in field [3:0].	RW	Yes	0
31:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.15 ACS Extended Capability Registers (Offsets 520h – 52Ch)

This section details the ACS Extended Capability registers. [Table 14-28](#) defines the register map.

**Table 14-28. 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	
<i>Reserved</i> (Upstream)			
Next Capability Offset (950h) (Downstream)	Capability Version (1h) (Downstream)	PCI Express Extended Capability ID (000Dh) (Downstream)	
<i>Reserved</i> (Upstream)			
ACS Control (Downstream)	ACS Capability (Downstream)		
<i>Reserved</i> (Upstream)			
<i>Reserved</i> (Downstream)	Egress Control Vector (Downstream)		
<i>Reserved</i>			

**Register 14-247. 520h ACS Extended Capability Header  
(Downstream Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> 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, the ACS Extended Capability is excluded from the Linked List of PCI Express Extended Capabilities; therefore, the upstream Port's register is effectively hidden from system software, and the non-zero value has no significant consequence.					
15:0	<i>Reserved</i>	Upstream	RsvdP	No	0000h
	PCI Express Extended Capability ID	Downstream	RO	Yes	000Dh
19:16	<i>Reserved</i>	Upstream	RsvdP	No	0h
	Capability Version	Downstream	RO	Yes	1h
31:20	<i>Reserved</i>	Upstream	RsvdP	No	000h
	Next Capability Offset Program to 950h, which addresses the <b>Vendor-Specific Extended Capability 2</b> structure.	Downstream	RO	Yes	950h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-248. 524h ACS Control and Capability  
(All Transparent Downstream Ports; Reserved (RsvdP) for the Upstream Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Encodings 01h through FFh directly indicate the number of each downstream Port's <b>Egress Control Vector</b> register <i>Peer-to-Peer Port x Control</i> bit (Downstream Ports, offset 528h[15:0]).  <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	HwInit	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-248. 524h ACS Control and Capability  
(All Transparent Downstream Ports; Reserved (RsvdP) for the Upstream Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	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	<b>Reserved</b>	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	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	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	<b>Reserved</b>	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	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-249. 528h Egress Control Vector  
(All Downstream Ports; Reserved (RsvdP) for the Upstream Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the ACS Control register ACS P2P Egress Control Enable bit (Downstream Ports, offset 524h[21]) is Set.					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 0 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 1 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 2 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 3 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 4 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 5 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 6 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
7	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 7 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-249. 528h Egress Control Vector  
(All Downstream Ports; Reserved (RsvdP) for the Upstream Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 8 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
9	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 9 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
10	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 10 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
11	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 11 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
12	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 12 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
13	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 13 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
14	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 14 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
15	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Peer-to-Peer Port 15 Control</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	Downstream	RW	Yes	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.16 Device-Specific Registers (Offsets 530h – F8Ch)

This section details the Device-Specific registers located at offsets 530h through F8Ch. Device-Specific registers are unique to the PEX 8619 and not referenced in the *PCI Express Base r2.0*. Table 14-29 defines the register map.

Other Device-Specific registers are detailed in [Section 14.14, “Device-Specific Registers \(Offsets 1C0h – 51Ch\).”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 14-29. Device-Specific Register Map (Offsets 530h – F8Ch)**

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	
		530h
		...
		540h
		544h
		...
		554h
		Factory Test Only 558h – 570h
		574h
		...
		628h
		62Ch
		...
		65Ch
		660h
		...
		67Ch
		680h
		...
		6BCh
		6C0h
		...
		73Ch
		Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h)
		Device-Specific Registers – Read Pacing (Offsets 544h – 554h)
		Device-Specific Registers – Port Configuration (Offsets 574h – 628h)
		Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch)
		Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)
		Device-Specific Registers – IOCAM Base and Limit Upper 16 Bits (Offsets 680h – 6BCh)
		Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-29. Device-Specific Register Map  
(Offsets 530h – F8Ch) (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															
Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch)																740h	...	...	83Ch																											
<i>Reserved</i>																840h –	93Ch																													
Device-Specific Registers – Ingress Credit Handler Port Pool (Offsets 940h – 94Ch)																940h	...	...	94Ch																											
Next Capability Offset 2 (000h)		1h		PCI Express Extended Capability ID 2 (000Bh)													950h	...	...	95Ch																										
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)																95Ch	...	...	95Ch																											
<i>Factory Test Only</i>																960h –	974h			978h																										
Device-Specific Registers – DMA Port Credit (Offsets 978h – 97Ch)																97Ch	...	...	980h																											
Device-Specific Registers – ACS Extended Capability (Offsets 980h – 9FCh)																9FCh	...	...	A00h																											
Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch)																B7Ch	...	...	B80h																											
Device-Specific Registers – Physical Layer (Offsets B80h – C30h)																C30h	...	...																												
<i>Reserved</i>																C34h –	DFCh			E00h																										
Device-Specific Registers – Port Configuration Header (Offsets E00h – E3Ch)																E3Ch	...	...																												
<i>Reserved</i>																E40h –	EECh			E40h																										
Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)																EFCh	...	...	F00h																											
Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch)																F3Ch	...	...	F40h																											
Device-Specific Registers – Error Reporting (Offsets F40h – F4Ch)																F4Ch	...	...	F50h																											
Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch)																F8Ch	...	...																												

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 14.16.1 Device-Specific Registers – Virtual Channel Weighted Round Robin Arbitration (Offsets 530h – 540h)

This section details the Device-Specific Virtual Channel (VC) Weighted Round-Robin (WRR) Arbitration registers. Table 14-30 defines the register map.

**Table 14-30. Device-Specific VC WRR Arbitration Register Map (Upstream 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 3 Weighted Round Robin Select	Port 2 Weighted Round Robin Select	Port 1 Weighted Round Robin Select	Port 0 Weighted Round Robin Select	530h
Port 7 Weighted Round Robin Select	Port 6 Weighted Round Robin Select	Port 5 Weighted Round Robin Select	Port 4 Weighted Round Robin Select	534h
Port 11 Weighted Round Robin Select	Port 10 Weighted Round Robin Select	Port 9 Weighted Round Robin Select	Port 8 Weighted Round Robin Select	538h
Port 15 Weighted Round Robin Select	Port 14 Weighted Round Robin Select	Port 13 Weighted Round Robin Select	Port 12 Weighted Round Robin Select	53Ch
VC Fixed Arbitration Select				540h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-250. 530h VC WRR 0**  
**(Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
For each Port, there are eight time slices to be divided amongst VC0 and VC1. A value of 0 indicates that VC0 has the “right of way,” and similarly, a value of 1 indicates that VC1 has the “right of way.” The default value, AAh, represents Round-Robin arbitration (equal access to each Port) for VC0 and VC1.				
7:0	<b>Port 0 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
15:8	<b>Port 1 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
23:16	<b>Port 2 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
31:24	<b>Port 3 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh

**Register 14-251. 534h VC WRR 1**  
**(Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
For each Port, there are eight time slices to be divided amongst VC0 and VC1. A value of 0 indicates that VC0 has the “right of way,” and similarly, a value of 1 indicates that VC1 has the “right of way.” The default value, AAh, represents Round-Robin arbitration (equal access to each Port) for VC0 and VC1.				
7:0	<b>Port 4 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
15:8	<b>Port 5 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
23:16	<b>Port 6 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
31:24	<b>Port 7 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-252. 538h VC WRR 2  
(Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
For each Port, there are eight time slices to be divided amongst VC0 and VC1. A value of 0 indicates that VC0 has the “right of way,” and similarly, a value of 1 indicates that VC1 has the “right of way.” The default value, AAh, represents Round-Robin arbitration (equal access to each Port) for VC0 and VC1.				
7:0	<b>Port 8 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
15:8	<b>Port 9 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
23:16	<b>Port 10 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
31:24	<b>Port 11 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh

**Register 14-253. 53Ch VC WRR 3  
(Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
For each Port, there are eight time slices to be divided amongst VC0 and VC1. A value of 0 indicates that VC0 has the “right of way,” and similarly, a value of 1 indicates that VC1 has the “right of way.” The default value, AAh, represents Round-Robin arbitration (equal access to each Port) for VC0 and VC1.				
7:0	<b>Port 12 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
15:8	<b>Port 13 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
23:16	<b>Port 14 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh
31:24	<b>Port 15 Weighted Round Robin Select</b> 0 = VC0 1 = VC1	RW	Yes	AAh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-254. 540h VC Fixed Arbitration Select  
(Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
1	<b>Port 1 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
2	<b>Port 2 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
3	<b>Port 3 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
4	<b>Port 4 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
5	<b>Port 5 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
6	<b>Port 6 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
7	<b>Port 7 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-254. 540h VC Fixed Arbitration Select  
(Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
9	<b>Port 9 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
10	<b>Port 10 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
11	<b>Port 11 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
12	<b>Port 12 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
13	<b>Port 13 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
14	<b>Port 14 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
15	<b>Port 15 VC0 High Priority</b> 0 = VC1 is given higher priority over VC0 when fixed VC priority is selected 1 = VC0 is given higher priority over VC1 when Fixed VC Priority is selected	RW	Yes	0
17:16	<b>DMA Arbitration Priority</b> 00b = Normal has priority 01b = DMA has priority 10b = Round-robin 11b = <b>Reserved</b>	RW	Yes	00b
18	<b>VC1 Shares VC0 Port Arbitration Table</b> 0 = VC1 uses the default Port Arbitration table 1 = VC1 on the upstream Port uses the same, possibly programmed, table that VC0 uses	RW	Yes	0
31:19	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.16.2 Device-Specific Registers – Read Pacing (Offsets 544h – 554h)

**Caution:** *Source Queuing and Read Pacing should not be concurrently enabled.  
The two features are incompatible and doing so can result in Fatal errors.*

This section details the Device-Specific Read Pacing registers. Although the Read Pacing feature is supported on all Ports, its registers are implemented only on the upstream Port; otherwise, these registers are **reserved**. [Table 14-31](#) defines the register map.

Read Pacing is described, in detail, in [Section 9.6, “Read Pacing.”](#)

**Table 14-31. Device-Specific Read Pacing Register Map  
(Upstream Port, and also the NT Port Virtual Interface  
if the NT Port is a Legacy NT 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	
<b>Reserved</b> (Upstream)	<b>Read Pacing Control</b> (Upstream)	544h
<b>Reserved</b> (Downstream)		
<b>Reserved</b> (Upstream)	<b>Read Spreading Control</b> (Upstream)	548h
<b>Reserved</b> (Downstream)		
<b>Read Pacing Threshold 0</b> (Upstream)		54Ch
<b>Reserved</b> (Downstream)		
<b>Read Pacing Threshold 1</b> (Upstream)		550h
<b>Reserved</b> (Downstream)		
<b>Reserved</b> (Upstream)	<b>Read Counter Clear</b> (Upstream)	554h
<b>Reserved</b> (Downstream)		

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-255. 544h Read Pacing Control****(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Caution:</b> Read Pacing and Source Queuing should not be concurrently enabled. The two features are incompatible and doing so can result in Fatal errors.</i>					
0	<b>Port 0 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 0 1 = Read Pacing is disabled for Port 0	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
1	<b>Port 1 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 1 1 = Read Pacing is disabled for Port 1	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
2	<b>Port 2 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 2 1 = Read Pacing is disabled for Port 2	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
3	<b>Port 3 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 3 1 = Read Pacing is disabled for Port 3	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
4	<b>Port 4 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 4 1 = Read Pacing is disabled for Port 4	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
5	<b>Port 5 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 5 1 = Read Pacing is disabled for Port 5	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
6	<b>Port 6 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 6 1 = Read Pacing is disabled for Port 6	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
7	<b>Port 7 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 7 1 = Read Pacing is disabled for Port 7	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-255. 544h Read Pacing Control**

(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port) (Cont.)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 8 1 = Read Pacing is disabled for Port 8	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
9	<b>Port 9 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 9 1 = Read Pacing is disabled for Port 9	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
10	<b>Port 10 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 10 1 = Read Pacing is disabled for Port 10	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
11	<b>Port 11 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 11 1 = Read Pacing is disabled for Port 11	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
12	<b>Port 12 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 12 1 = Read Pacing is disabled for Port 12	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
13	<b>Port 13 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 13 1 = Read Pacing is disabled for Port 13	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
14	<b>Port 14 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 14 1 = Read Pacing is disabled for Port 14	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
15	<b>Port 15 Read Pacing Disable</b> 0 = Read Pacing is enabled for Port 15 1 = Read Pacing is disabled for Port 15	Upstream	RW	Yes	1
	<b>Reserved</b>	Downstream	RsvdP	No	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-256. 548h Read Spreading Control****(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Caution:</b> Memory Read Spreading and Source Queuing should not be concurrently enabled. The two features are incompatible and doing so can result in Fatal errors.</i>					
<i><b>Note:</b> Read Pacing (Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port, offset 544h) must be enabled for Read Spreading to be enabled. (That is, for a Port to have Read Spreading enabled, the corresponding Port bits in offset 544h and this register must both be Cleared.)</i>					
0	<b>Port 0 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 0 1 = Memory Read Spreading is disabled for Port 0	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
1	<b>Port 1 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 1 1 = Memory Read Spreading is disabled for Port 1	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
2	<b>Port 2 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 2 1 = Memory Read Spreading is disabled for Port 2	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
3	<b>Port 3 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 3 1 = Memory Read Spreading is disabled for Port 3	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
4	<b>Port 4 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 4 1 = Memory Read Spreading is disabled for Port 4	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
5	<b>Port 5 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 5 1 = Memory Read Spreading is disabled for Port 5	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
6	<b>Port 6 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 6 1 = Memory Read Spreading is disabled for Port 6	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
7	<b>Port 7 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 7 1 = Memory Read Spreading is disabled for Port 7	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-256. 548h Read Spreading Control  
(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 8 1 = Memory Read Spreading is disabled for Port 8	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
9	<b>Port 9 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 9 1 = Memory Read Spreading is disabled for Port 9	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
10	<b>Port 10 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 10 1 = Memory Read Spreading is disabled for Port 10	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
11	<b>Port 11 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 11 1 = Memory Read Spreading is disabled for Port 11	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
12	<b>Port 12 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 12 1 = Memory Read Spreading is disabled for Port 12	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
13	<b>Port 13 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 13 1 = Memory Read Spreading is disabled for Port 13	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
14	<b>Port 14 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 14 1 = Memory Read Spreading is disabled for Port 14	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
15	<b>Port 15 Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for Port 15 1 = Memory Read Spreading is disabled for Port 15	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-257. 54Ch Read Pacing Threshold 0**

(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12:0	<b>MRd_Size_Outstanding_Threshold for x8 Port</b> Memory Read size outstanding threshold for x8 Ports. This Counter specifies the maximum DWord of Memory Read Requests that can be outstanding for a x8 Port. Default value of 600h programs the threshold to 6 KB.	Upstream	RW	Yes	600h
	<b>Reserved</b>	Downstream	RsvdP	No	000h
15:13	<b>Reserved</b>		RsvdP	No	000b
28:16	<b>MRd_Size_Outstanding_Threshold for x4 Port</b> Memory Read size outstanding threshold for x4 Ports. This Counter specifies the maximum DWord of Memory Read Requests that can be outstanding for a x4 Port. Default value of 400h programs the threshold to 4 KB.	Upstream	RW	Yes	400h
	<b>Reserved</b>	Downstream	RsvdP	No	000h
31:29	<b>Reserved</b>		RsvdP	No	000b

**Register 14-258. 550h Read Pacing Threshold 1**

(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12:0	<b>MRd_Size_Outstanding_Threshold for x1 Port</b> Memory Read size outstanding threshold for x1 Ports. This Counter specifies the maximum DWord of Memory Read Requests that can be outstanding for a x1 Port. Default value of 100h programs the threshold to 1 KB.	Upstream	RW	Yes	100h
	<b>Reserved</b>	Downstream	RsvdP	No	000h
31:13	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-259. 554h Read Counter Clear**

(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 0. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 0.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
1	<b>Port 1 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 1. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 1.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
2	<b>Port 2 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 2. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 2.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
3	<b>Port 3 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 3. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 3.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
4	<b>Port 4 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 4. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 4.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
5	<b>Port 5 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 5. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 5.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
6	<b>Port 6 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 6. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 6.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				
7	<b>Port 7 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 7. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 7.	Upstream	RW1C	Yes	0
	<b>Reserved</b>				

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-259. 554h Read Counter Clear**

(Upstream Port, and also the NT Port Virtual Interface if the NT Port is a Legacy NT Port) (Cont.)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 8. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 8.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
9	<b>Port 9 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 9. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 9.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
10	<b>Port 10 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 10. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 10.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
11	<b>Port 11 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 11. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 11.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
12	<b>Port 12 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 12. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 12.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
13	<b>Port 13 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 13. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 13.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
14	<b>Port 14 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 14. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 14.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
15	<b>Port 15 Reset_Outstanding_MRd_Size_Counter</b> Reset Outstanding Memory Read Size Counter for Port 15. Writing 1 to this bit resets the Outstanding Memory Read Request Size Counter for Port 15.	Upstream	RW1C	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.3 Device-Specific Registers – Port Configuration (Offsets 574h – 628h)

This section details the Device-Specific Port Configuration register. [Table 14-32](#) defines the register map.

**Table 14-32. Device-Specific Port Configuration Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	574h
		<i>Factory Test Only</i>	578h – 628h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-260. 574h Port Configuration****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The configurations for each Port are provided in <a href="#">Table 14-33</a> .				
3:0	<b>Port Configuration</b> Port Configuration Link width, per Port. The serial EEPROM bit values always override the <b>STRAP_PORTCFG[3:0]</b> Strapping inputs (if the serial EEPROM values are loaded; refer to <a href="#">Table 14-33</a> ). This register is reset only by a Fundamental Reset ( <b>PEX_PERST#</b> assertion). 0h = x1, x1 1h = x4, x1, x1 2h = x4, x4, x1, x1 3h = x4, x4, x4, x1, x1, x1, x1 4h = x4, x4, x4, x4 5h = x8, x1, x1, x1, x1, x1, x1, x1 6h = x8, x4, x1, x1, x1, x1 7h = x8, x4, x4 8h = x8, x8 All other encodings are <i>reserved</i> .	RO	Yes	Set by STRAP_PORTCFG[3:0] input levels
15:4	<b>Reserved</b>	RsvdP	No	000h
19:16	<b>Force Native x2 Width</b> Force native x2 width to Ports 0, 1, 2, and 3 for standard Port configurations of x4 or greater. Initializes with serial EEPROM (if present).	RO	Serial EEPROM Only	0h
31:20	<b>Reserved</b>	RsvdP	No	000h

**Table 14-33. Port Configurations**

STRAP_PORTCFG [3:0]	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15
(default) 0h	x1	x1	x1	x1	x1	x1										
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	x1
2h	x4	x4	x1	x1							x1	x1	x1	x1	x1	x1
3h	x4	x4	x4	x1								x1		x1		x1
4h	x4	x4	x4	x4												
5h	x8	x1		x1		x1		x1								
6h	x8	x4		x1							x1		x1		x1	
7h	x8	x4		x4												
8h	x8	x8														

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

#### 14.16.4 Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch)

This section details the Device-Specific General-Purpose Input/Output (GPIO) Configuration, Status, and Control registers. Table 14-34 defines the register map.

**Table 14-34. Device-Specific GPIO Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
	GPIO 0_15 Direction Control	62Ch
	GPIO 16_31 Direction Control	630h
	<i>Reserved</i>	634h
	GPIO 0_31 Input De-Bounce	638h
<i>Reserved</i>	GPIO 0_15 Input Data	63Ch
<i>Reserved</i>	GPIO 16_31 Input Data	640h
<i>Reserved</i>	GPIO 0_15 Output Data	644h
<i>Reserved</i>	GPIO 16_31 Output Data	648h
	GPIO 0_31 Interrupt Polarity	64Ch
	GPIO 0_31 Interrupt Status	650h
	GPIO 0_31 Interrupt Mask	654h
	<i>Factory Test Only</i>	658h – 65Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-261. 62Ch GPIO 0\_15 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO0 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO0 Input Data</b> register (offset 63Ch[0]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO0 Output Data</b> register (offset 644h[0]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 0	RWS	Yes	0
1	<b>GPIO0 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
2	<b>GPIO1 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO1 Input Data</b> register (offset 63Ch[1]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO1 Output Data</b> register (offset 644h[1]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 1	RWS	Yes	0
3	<b>GPIO1 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
4	<b>GPIO2 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO2 Input Data</b> register (offset 63Ch[2]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO2 Output Data</b> register (offset 644h[2]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 2	RWS	Yes	0
5	<b>GPIO2 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
6	<b>GPIO3 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO3 Input Data</b> register (offset 63Ch[3]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO3 Output Data</b> register (offset 644h[3]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 3	RWS	Yes	0
7	<b>GPIO3 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-261. 62Ch GPIO 0\_15 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO4 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO4 Input Data</b> register (offset 63Ch[4]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO4 Output Data</b> register (offset 644h[4]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 4	RWS	Yes	0
9	<b>GPIO4 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
10	<b>GPIO5 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO5 Input Data</b> register (offset 63Ch[5]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO5 Output Data</b> register (offset 644h[5]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 5	RWS	Yes	0
11	<b>GPIO5 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
12	<b>GPIO6 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO6 Input Data</b> register (offset 63Ch[6]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO6 Output Data</b> register (offset 644h[6]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 6	RWS	Yes	0
13	<b>GPIO6 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
14	<b>GPIO7 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO7 Input Data</b> register (offset 63Ch[7]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO7 Output Data</b> register (offset 644h[7]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 7	RWS	Yes	0
15	<b>GPIO7 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-261. 62Ch GPIO 0\_15 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>GPIO8 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO8 Input Data</b> register (offset 63Ch[8]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO8 Output Data</b> register (offset 644h[8]) 1 = Serial Hot Plug <b>PERST#</b> output for Hot Plug Virtual Port 8	RWS	Yes	0
17	<b>GPIO8 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
18	<b>GPIO9 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO9 Input Data</b> register (offset 63Ch[9]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO9 Output Data</b> register (offset 644h[9]) 1 = Serial Hot Plug <b>PERST#</b> output for Hot Plug Virtual Port 9	RWS	Yes	0
19	<b>GPIO9 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
20	<b>GPIO10 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO10 Input Data</b> register (offset 63Ch[10]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO10 Output Data</b> register (offset 644h[10]) 1 = Serial Hot Plug <b>PERST#</b> output for Hot Plug Virtual Port 10	RWS	Yes	0
21	<b>GPIO10 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
22	<b>GPIO11 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO11 Input Data</b> register (offset 63Ch[11]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO11 Output Data</b> register (offset 644h[11]) 1 = Serial Hot Plug <b>PERST#</b> output for Hot Plug Virtual Port 11	RWS	Yes	0
23	<b>GPIO11 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-261. 62Ch GPIO 0\_15 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>GPIO12 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO12 Input Data</b> register (offset 63Ch[12]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO12 Output Data</b> register (offset 644h[12]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 12	RWS	Yes	0
25	<b>GPIO12 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
26	<b>GPIO13 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO13 Input Data</b> register (offset 63Ch[13]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO13 Output Data</b> register (offset 644h[13]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 13	RWS	Yes	0
27	<b>GPIO13 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
28	<b>GPIO14 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO14 Input Data</b> register (offset 63Ch[14]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO14 Output Data</b> register (offset 644h[14]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 14	RWS	Yes	0
29	<b>GPIO14 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
30	<b>GPIO15 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO15 Input Data</b> register (offset 63Ch[15]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO15 Output Data</b> register (offset 644h[15]) 1 = Serial Hot Plug PERST# output for Hot Plug Virtual Port 15	RWS	Yes	0
31	<b>GPIO15 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-262. 630h GPIO 16\_31 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO16 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO16 Input Data</b> register (offset 640h[0]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO16 Output Data</b> register (offset 648h[0]) 1 = <i>Reserved</i>	RWS	Yes	0
1	<b>GPIO16 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
2	<b>GPIO17 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO17 Input Data</b> register (offset 640h[1]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO17 Output Data</b> register (offset 648h[1]) 1 = <i>Reserved</i>	RWS	Yes	0
3	<b>GPIO17 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
4	<b>GPIO18 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO18 Input Data</b> register (offset 640h[2]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO18 Output Data</b> register (offset 648h[2]) 1 = <i>Reserved</i>	RWS	Yes	0
5	<b>GPIO18 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
6	<b>GPIO19 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO19 Input Data</b> register (offset 640h[3]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO19 Output Data</b> register (offset 648h[3]) 1 = <i>Reserved</i>	RWS	Yes	0
7	<b>GPIO19 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-262. 630h GPIO 16\_31 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO20 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO20 Input Data</b> register (offset 640h[4]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO20 Output Data</b> register (offset 648h[4]) 1 = <i>Reserved</i>	RWS	Yes	0
9	<b>GPIO20 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
10	<b>GPIO21 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO21 Input Data</b> register (offset 640h[5]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO21 Output Data</b> register (offset 648h[5]) 1 = <i>Reserved</i>	RWS	Yes	0
11	<b>GPIO21 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
12	<b>GPIO22 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO22 Input Data</b> register (offset 640h[6]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO22 Output Data</b> register (offset 648h[6]) 1 = <i>Reserved</i>	RWS	Yes	0
13	<b>GPIO22 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
14	<b>GPIO23 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO23 Input Data</b> register (offset 640h[7]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO23 Output Data</b> register (offset 648h[7]) 1 = <i>Reserved</i>	RWS	Yes	0
15	<b>GPIO23 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-262. 630h GPIO 16\_31 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>GPIO24 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO24 Input Data</b> register (offset 640h[8]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO24 Output Data</b> register (offset 648h[8]) 1 = <b>Reserved</b>	RWS	Yes	0
17	<b>GPIO24 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
18	<b>GPIO25 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO25 Input Data</b> register (offset 640h[9]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO25 Output Data</b> register (offset 648h[9]) 1 = <b>Reserved</b>	RWS	Yes	0
19	<b>GPIO25 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
20	<b>GPIO26 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO26 Input Data</b> register (offset 640h[10]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO26 Output Data</b> register (offset 648h[10]) 1 = <b>Reserved</b>	RWS	Yes	0
21	<b>GPIO26 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
22	<b>GPIO27 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO27 Input Data</b> register (offset 640h[11]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO27 Output Data</b> register (offset 648h[11]) 1 = <b>Reserved</b>	RWS	Yes	0
23	<b>GPIO27 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-262. 630h GPIO 16\_31 Direction Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>GPIO28 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO28 Input Data</b> register (offset 640h[12]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO28 Output Data</b> register (offset 648h[12]) 1 = <b>Reserved</b>	RWS	Yes	0
25	<b>GPIO28 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
26	<b>GPIO29 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO29 Input Data</b> register (offset 640h[13]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO29 Output Data</b> register (offset 648h[13]) 1 = <b>Reserved</b>	RWS	Yes	0
27	<b>GPIO29 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
28	<b>GPIO30 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO30 Input Data</b> register (offset 640h[14]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO30 Output Data</b> register (offset 648h[14]) 1 = <b>Reserved</b>	RWS	Yes	0
29	<b>GPIO30 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
30	<b>GPIO31 Source/Destination</b> <b>As Input:</b> 0 = To <b>GPIO31 Input Data</b> register (offset 640h[15]) 1 = General interrupt (INTx, MSI, or PEX_INTA#) <b>As Output:</b> 0 = From <b>GPIO31 Output Data</b> register (offset 648h[15]) 1 = <b>Reserved</b>	RWS	Yes	0
31	<b>GPIO31 Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-263. 638h GPIO 0\_31 Input De-Bounce  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<p><b>GPIOx Input De-Bounce Control</b> Controls de-bounce when the corresponding GPIOx signal is configured as an input (offset 62Ch[x], is Cleared, where x is the bit number associated with that GPIOx signal). Bits [31:0] correspond to GPIO[31:0], respectively.</p> <p>0 = GPIOx input is not de-bounced 1 = GPIOx input is de-bounced; de-bounce time is approximately 1.3 ms</p>	RWS	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-264. 63Ch GPIO 0\_15 Input Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO0 Input Data</b> If GPIO0 is configured as an output (offset 62Ch[1], is Set), Reads return 0. If GPIO0 is configured as an input (offset 62Ch[1], is Cleared), Reads return the logic value of the voltage on GPIO0.	RO	No	0
1	<b>GPIO1 Input Data</b> If GPIO1 is configured as an output (offset 62Ch[3], is Set), Reads return 0. If GPIO1 is configured as an input (offset 62Ch[3], is Cleared), Reads return the logic value of the voltage on GPIO1.	RO	No	0
2	<b>GPIO2 Input Data</b> If GPIO2 is configured as an output (offset 62Ch[5], is Set), Reads return 0. If GPIO2 is configured as an input (offset 62Ch[5], is Cleared), Reads return the logic value of the voltage on GPIO2.	RO	No	0
3	<b>GPIO3 Input Data</b> If GPIO3 is configured as an output (offset 62Ch[7], is Set), Reads return 0. If GPIO3 is configured as an input (offset 62Ch[7], is Cleared), Reads return the logic value of the voltage on GPIO3.	RO	No	0
4	<b>GPIO4 Input Data</b> If GPIO4 is configured as an output (offset 62Ch[9], is Set), Reads return 0. If GPIO4 is configured as an input (offset 62Ch[9], is Cleared), Reads return the logic value of the voltage on GPIO4.	RO	No	0
5	<b>GPIO5 Input Data</b> If GPIO5 is configured as an output (offset 62Ch[11], is Set), Reads return 0. If GPIO5 is configured as an input (offset 62Ch[11], is Cleared), Reads return the logic value of the voltage on GPIO5.	RO	No	0
6	<b>GPIO6 Input Data</b> If GPIO6 is configured as an output (offset 62Ch[13], is Set), Reads return 0. If GPIO6 is configured as an input (offset 62Ch[13], is Cleared), Reads return the logic value of the voltage on GPIO6.	RO	No	0
7	<b>GPIO7 Input Data</b> If GPIO7 is configured as an output (offset 62Ch[15], is Set), Reads return 0. If GPIO7 is configured as an input (offset 62Ch[15], is Cleared), Reads return the logic value of the voltage on GPIO7.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-264. 63Ch GPIO 0\_15 Input Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO8 Input Data</b> If GPIO8 is configured as an output (offset 62Ch[17], is Set), Reads return 0. If GPIO8 is configured as an input (offset 62Ch[17], is Cleared), Reads return the logic value of the voltage on GPIO8.	RO	No	0
9	<b>GPIO9 Input Data</b> If GPIO9 is configured as an output (offset 62Ch[19], is Set), Reads return 0. If GPIO9 is configured as an input (offset 62Ch[19], is Cleared), Reads return the logic value of the voltage on GPIO9.	RO	No	0
10	<b>GPIO10 Input Data</b> If GPIO10 is configured as an output (offset 62Ch[21], is Set), Reads return 0. If GPIO10 is configured as an input (offset 62Ch[21], is Cleared), Reads return the logic value of the voltage on GPIO10.	RO	No	0
11	<b>GPIO11 Input Data</b> If GPIO11 is configured as an output (offset 62Ch[23], is Set), Reads return 0. If GPIO11 is configured as an input (offset 62Ch[23], is Cleared), Reads return the logic value of the voltage on GPIO11.	RO	No	0
12	<b>GPIO12 Input Data</b> If GPIO12 is configured as an output (offset 62Ch[25], is Set), Reads return 0. If GPIO12 is configured as an input (offset 62Ch[25], is Cleared), Reads return the logic value of the voltage on GPIO12.	RO	No	0
13	<b>GPIO13 Input Data</b> If GPIO13 is configured as an output (offset 62Ch[27], is Set), Reads return 0. If GPIO13 is configured as an input (offset 62Ch[27], is Cleared), Reads return the logic value of the voltage on GPIO13.	RO	No	0
14	<b>GPIO14 Input Data</b> If GPIO14 is configured as an output (offset 62Ch[29], is Set), Reads return 0. If GPIO14 is configured as an input (offset 62Ch[29], is Cleared), Reads return the logic value of the voltage on GPIO14.	RO	No	0
15	<b>GPIO15 Input Data</b> If GPIO15 is configured as an output (offset 62Ch[31], is Set), Reads return 0. If GPIO15 is configured as an input (offset 62Ch[31], is Cleared), Reads return the logic value of the voltage on GPIO15.	RO	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-265. 640h GPIO 16\_31 Input Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO16 Input Data</b> If GPIO16 is configured as an output (offset 630h[1], is Set), Reads return 0. If GPIO16 is configured as an input (offset 630h[1], is Cleared), Reads return the logic value of the voltage on GPIO16.	RO	No	0
1	<b>GPIO17 Input Data</b> If GPIO17 is configured as an output (offset 630h[3], is Set), Reads return 0. If GPIO17 is configured as an input (offset 630h[3], is Cleared), Reads return the logic value of the voltage on GPIO17.	RO	No	0
2	<b>GPIO18 Input Data</b> If GPIO18 is configured as an output (offset 630h[5], is Set), Reads return 0. If GPIO18 is configured as an input (offset 630h[5], is Cleared), Reads return the logic value of the voltage on GPIO18.	RO	No	0
3	<b>GPIO19 Input Data</b> If GPIO19 is configured as an output (offset 630h[7], is Set), Reads return 0. If GPIO19 is configured as an input (offset 630h[7], is Cleared), Reads return the logic value of the voltage on GPIO19.	RO	No	0
4	<b>GPIO20 Input Data</b> If GPIO20 is configured as an output (offset 630h[9], is Set), Reads return 0. If GPIO20 is configured as an input (offset 630h[9], is Cleared), Reads return the logic value of the voltage on GPIO20.	RO	No	0
5	<b>GPIO21 Input Data</b> If GPIO21 is configured as an output (offset 630h[11], is Set), Reads return 0. If GPIO21 is configured as an input (offset 630h[11], is Cleared), Reads return the logic value of the voltage on GPIO21.	RO	No	0
6	<b>GPIO22 Input Data</b> If GPIO22 is configured as an output (offset 630h[13], is Set), Reads return 0. If GPIO22 is configured as an input (offset 630h[13], is Cleared), Reads return the logic value of the voltage on GPIO22.	RO	No	0
7	<b>GPIO23 Input Data</b> If GPIO23 is configured as an output (offset 630h[15], is Set), Reads return 0. If GPIO23 is configured as an input (offset 630h[15], is Cleared), Reads return the logic value of the voltage on GPIO23.	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-265. 640h GPIO 16\_31 Input Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO24 Input Data</b> If GPIO24 is configured as an output (offset 630h[17], is Set), Reads return 0. If GPIO24 is configured as an input (offset 630h[17], is Cleared), Reads return the logic value of the voltage on GPIO24.	RO	No	0
9	<b>GPIO25 Input Data</b> If GPIO25 is configured as an output (offset 630h[19], is Set), Reads return 0. If GPIO25 is configured as an input (offset 630h[19], is Cleared), Reads return the logic value of the voltage on GPIO25.	RO	No	0
10	<b>GPIO26 Input Data</b> If GPIO26 is configured as an output (offset 630h[21], is Set), Reads return 0. If GPIO26 is configured as an input (offset 630h[21], is Cleared), Reads return the logic value of the voltage on GPIO26.	RO	No	0
11	<b>GPIO27 Input Data</b> If GPIO27 is configured as an output (offset 630h[23], is Set), Reads return 0. If GPIO27 is configured as an input (offset 630h[23], is Cleared), Reads return the logic value of the voltage on GPIO27.	RO	No	0
12	<b>GPIO28 Input Data</b> If GPIO28 is configured as an output (offset 630h[25], is Set), Reads return 0. If GPIO28 is configured as an input (offset 630h[25], is Cleared), Reads return the logic value of the voltage on GPIO28.	RO	No	0
13	<b>GPIO29 Input Data</b> If GPIO29 is configured as an output (offset 630h[27], is Set), Reads return 0. If GPIO29 is configured as an input (offset 630h[27], is Cleared), Reads return the logic value of the voltage on GPIO29.	RO	No	0
14	<b>GPIO30 Input Data</b> If GPIO30 is configured as an output (offset 630h[29], is Set), Reads return 0. If GPIO30 is configured as an input (offset 630h[29], is Cleared), Reads return the logic value of the voltage on GPIO30.	RO	No	0
15	<b>GPIO31 Input Data</b> If GPIO31 is configured as an output (offset 630h[31], is Set), Reads return 0. If GPIO31 is configured as an input (offset 630h[31], is Cleared), Reads return the logic value of the voltage on GPIO31.	RO	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-266. 644h GPIO 0\_15 Output Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO0 Output Data</b> If GPIO0 is configured as an output (offset 62Ch[1], is Set), the value written to this bit is immediately driven to the GPIO0 output. Reads return the value written.	RWS	Yes	0
1	<b>GPIO1 Output Data</b> If GPIO1 is configured as an output (offset 62Ch[3], is Set), the value written to this bit is immediately driven to the GPIO1 output. Reads return the value written.	RWS	Yes	0
2	<b>GPIO2 Output Data</b> If GPIO2 is configured as an output (offset 62Ch[5], is Set), the value written to this bit is immediately driven to the GPIO2 output. Reads return the value written.	RWS	Yes	0
3	<b>GPIO3 Output Data</b> If GPIO3 is configured as an output (offset 62Ch[7], is Set), the value written to this bit is immediately driven to the GPIO3 output. Reads return the value written.	RWS	Yes	0
4	<b>GPIO4 Output Data</b> If GPIO4 is configured as an output (offset 62Ch[9], is Set), the value written to this bit is immediately driven to the GPIO4 output. Reads return the value written.	RWS	Yes	0
5	<b>GPIO5 Output Data</b> If GPIO5 is configured as an output (offset 62Ch[11], is Set), the value written to this bit is immediately driven to the GPIO5 output. Reads return the value written.	RWS	Yes	0
6	<b>GPIO6 Output Data</b> If GPIO6 is configured as an output (offset 62Ch[13], is Set), the value written to this bit is immediately driven to the GPIO6 output. Reads return the value written.	RWS	Yes	0
7	<b>GPIO7 Output Data</b> If GPIO7 is configured as an output (offset 62Ch[15], is Set), the value written to this bit is immediately driven to the GPIO7 output. Reads return the value written.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-266. 644h GPIO 0\_15 Output Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO8 Output Data</b> If GPIO8 is configured as an output (offset 62Ch[17], is Set), the value written to this bit is immediately driven to the GPIO8 output. Reads return the value written.	RWS	Yes	0
9	<b>GPIO9 Output Data</b> If GPIO9 is configured as an output (offset 62Ch[19], is Set), the value written to this bit is immediately driven to the GPIO9 output. Reads return the value written.	RWS	Yes	0
10	<b>GPIO10 Output Data</b> If GPIO10 is configured as an output (offset 62Ch[21], is Set), the value written to this bit is immediately driven to the GPIO10 output. Reads return the value written.	RWS	Yes	0
11	<b>GPIO11 Output Data</b> If GPIO11 is configured as an output (offset 62Ch[23], is Set), the value written to this bit is immediately driven to the GPIO11 output. Reads return the value written.	RWS	Yes	0
12	<b>GPIO12 Output Data</b> If GPIO12 is configured as an output (offset 62Ch[25], is Set), the value written to this bit is immediately driven to the GPIO12 output. Reads return the value written.	RWS	Yes	0
13	<b>GPIO13 Output Data</b> If GPIO13 is configured as an output (offset 62Ch[27], is Set), the value written to this bit is immediately driven to the GPIO13 output. Reads return the value written.	RWS	Yes	0
14	<b>GPIO14 Output Data</b> If GPIO14 is configured as an output (offset 62Ch[29], is Set), the value written to this bit is immediately driven to the GPIO14 output. Reads return the value written.	RWS	Yes	0
15	<b>GPIO15 Output Data</b> If GPIO15 is configured as an output (offset 62Ch[31], is Set), the value written to this bit is immediately driven to the GPIO15 output. Reads return the value written.	RWS	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-267. 648h GPIO 16\_31 Output Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>GPIO16 Output Data</b> If GPIO16 is configured as an output (offset 630h[1], is Set), the value written to this bit is immediately driven to the GPIO16 output. Reads return the value written.	RWS	Yes	0
1	<b>GPIO17 Output Data</b> If GPIO17 is configured as an output (offset 630h[3], is Set), the value written to this bit is immediately driven to the GPIO17 output. Reads return the value written.	RWS	Yes	0
2	<b>GPIO18 Output Data</b> If GPIO18 is configured as an output (offset 630h[5], is Set), the value written to this bit is immediately driven to the GPIO18 output. Reads return the value written.	RWS	Yes	0
3	<b>GPIO19 Output Data</b> If GPIO19 is configured as an output (offset 630h[7], is Set), the value written to this bit is immediately driven to the GPIO19 output. Reads return the value written.	RWS	Yes	0
4	<b>GPIO20 Output Data</b> If GPIO20 is configured as an output (offset 630h[9], is Set), the value written to this bit is immediately driven to the GPIO20 output. Reads return the value written.	RWS	Yes	0
5	<b>GPIO21 Output Data</b> If GPIO21 is configured as an output (offset 630h[11], is Set), the value written to this bit is immediately driven to the GPIO21 output. Reads return the value written.	RWS	Yes	0
6	<b>GPIO22 Output Data</b> If GPIO22 is configured as an output (offset 630h[13], is Set), the value written to this bit is immediately driven to the GPIO22 output. Reads return the value written.	RWS	Yes	0
7	<b>GPIO23 Output Data</b> If GPIO23 is configured as an output (offset 630h[15], is Set), the value written to this bit is immediately driven to the GPIO23 output. Reads return the value written.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-267. 648h GPIO 16\_31 Output Data  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>GPIO24 Output Data</b> If GPIO24 is configured as an output (offset 630h[17], is Set), the value written to this bit is immediately driven to the GPIO24 output. Reads return the value written.	RWS	Yes	0
9	<b>GPIO25 Output Data</b> If GPIO25 is configured as an output (offset 630h[19], is Set), the value written to this bit is immediately driven to the GPIO25 output. Reads return the value written.	RWS	Yes	0
10	<b>GPIO26 Output Data</b> If GPIO26 is configured as an output (offset 630h[21], is Set), the value written to this bit is immediately driven to the GPIO26 output. Reads return the value written.	RWS	Yes	0
11	<b>GPIO27 Output Data</b> If GPIO27 is configured as an output (offset 630h[23], is Set), the value written to this bit is immediately driven to the GPIO27 output. Reads return the value written.	RWS	Yes	0
12	<b>GPIO28 Output Data</b> If GPIO28 is configured as an output (offset 630h[25], is Set), the value written to this bit is immediately driven to the GPIO28 output. Reads return the value written.	RWS	Yes	0
13	<b>GPIO29 Output Data</b> If GPIO29 is configured as an output (offset 630h[27], is Set), the value written to this bit is immediately driven to the GPIO29 output. Reads return the value written.	RWS	Yes	0
14	<b>GPIO30 Output Data</b> If GPIO30 is configured as an output (offset 630h[29], is Set), the value written to this bit is immediately driven to the GPIO30 output. Reads return the value written.	RWS	Yes	0
15	<b>GPIO31 Output Data</b> If GPIO31 is configured as an output (offset 630h[31], is Set), the value written to this bit is immediately driven to the GPIO31 output. Reads return the value written.	RWS	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-268. 64Ch GPIO 0\_31 Interrupt Polarity  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>GPIO Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the corresponding GPIOx signal. Bits [31:0] correspond to GPIO[31:0], respectively. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0000_0000h

**Register 14-269. 650h GPIO 0\_31 Interrupt Status  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

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 bits in the <a href="#">GPIO 0_31 Interrupt Mask</a> register (offset 654h).				
31:0	<b>GPIO Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the corresponding GPIOx signal. Bits [31:0] correspond to GPIO[31:0], respectively. 0 = GPIO interrupt is inactive. 1 = GPIO interrupt is active. Interrupt status remains Set, as long the corresponding GPIOx signal is asserted, and Clears on its own when the corresponding GPIOx input de-asserts to the inactive state.	RO	No	0000_0000h

**Register 14-270. 654h GPIO 0\_31 Interrupt Mask  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

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 bits in the <a href="#">GPIO 0_31 Interrupt Status</a> register (offset 650h).				
31:0	<b>GPIO Interrupt Mask</b> Indicates whether GPIO interrupts are masked or not masked for the corresponding GPIOx signal. Bits [31:0] correspond to GPIO[31:0], respectively. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register is not updated.	RWS	Yes	FFFF_FFFFh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 14.16.5 Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)

This section details the Device-Specific Ingress Control and Port Enable registers, which also include the **Negotiated Link Width**, **Port Cut-Thru Enable Status**, and **Ingress PLL RAM ECC 1-Bit Counter** registers. [Table 14-35](#) defines the register map.

**Table 14-35. Device-Specific Ingress Control and Port Enable 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	
		Ingress Control		660h
		Ingress Control Shadow		664h
	<i>Reserved</i>		Port Enable Status	668h
		Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7		66Ch
		Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15		670h
	<i>Reserved</i>		Port Cut-Thru Enable Status	674h
	<i>Reserved</i>		Ingress PLL RAM ECC 1-Bit Counter	678h
		Ingress Control 2		67Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-271. 660h Ingress Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
22:0	<b>Factory Test Only</b>	RWS	Yes	0-0h
23	<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
25:24	<b>Factory Test Only</b>	RWS	Yes	00b
26	<b>Disable Upstream Port BAR0 and BAR1</b> 0 = Enables the upstream Port <b>Base Address 0</b> and <b>Base Address 1</b> registers ( <b>BAR0</b> and <b>BAR1</b> , Upstream Port, offsets 10h and 14h, respectively) 1 = Disables upstream Port <b>BAR0</b> and <b>BAR1</b>	RWS	Yes	0
27	<b>Not used</b>	RWS	Yes	0
28	<b>Disable VGA BIOS Memory Access Decoding</b> 0 = Enables the <b>Bridge Control</b> register <b>VGA 16-Bit Decode Enable</b> , <b>VGA Enable</b> , and <b>ISA Enable</b> bits (offset 3Ch[20:18], respectively), and enables decoding of the PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are blocked) 1 = Disables the <b>Bridge Control</b> register <b>VGA 16-Bit Decode Enable</b> , <b>VGA Enable</b> , and <b>ISA Enable</b> bits (offset 3Ch[20:18], respectively), and disables decoding of the PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are <b>not</b> blocked)	RWS	Yes	1
30:29	<b>Factory Test Only</b>	RWS	Yes	00b
31	<b>Not used</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-272. 664h Ingress Control Shadow  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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
3:2	<b>Factory Test Only</b>	RWS	Yes	00b
4	<b>Drop EP TLPs</b> Drop Endpoint TLPs. 1 = Endpoint TLP was dropped	RWS	Yes	0
5	<b>No Special Treatment for Relaxed Ordering Traffic</b> The PEX 8619 supports Relaxed Ordering for Completions. By default, if the RO attribute is Set within a Completion, then 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. 1 = Device-Specific Relaxed Ordering Completion will not be flagged to the egress block	RWS	Yes	0
7:6	<b>Factory Test Only</b>	RWS	Yes	00b
8	<b>Drop ECRC TLPs</b> Drop End-to-end Cyclic Redundancy Check (ECRC) TLPs. 1 = ECRC TLP was dropped	RWS	Yes	0
10:9	<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
11	<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 in the <a href="#">Link Error Status Virtual</a> register <a href="#">Link Side Uncorrectable Error Message Drop Status</a> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <a href="#">FE0h[3]</a> ). If the corresponding <a href="#">Link Error Mask Virtual</a> register <a href="#">Link Side Uncorrectable Error Message Drop Mask</a> bit (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <a href="#">FE4h[3]</a> ) is Set, the NT Port Virtual Interface signals an Interrupt (INTx, MSI, or <a href="#">PEX_INTA#</a> ) to the Local Host through the upstream Port, if interrupts are enabled.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-272. 664h Ingress Control Shadow  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<b>Factory Test Only</b>	RWS	Yes	000b
15	<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
19:16	<b>Factory Test Only</b>	RWS	Yes	0h
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
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
31:30	<b>Not used</b>	RWS	Yes	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-273. 668h Port Enable Status  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The value of this register depends upon the Port configuration (refer to <a href="#">Table 14-36</a> for a sample mapping).				
0	<b>Port 0 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
1	<b>Port 1 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
2	<b>Port 2 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
3	<b>Port 3 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-273. 668h Port Enable Status  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 4 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
5	<b>Port 5 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
6	<b>Port 6 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
7	<b>Port 7 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-273. 668h Port Enable Status  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
9	<b>Port 9 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
10	<b>Port 10 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
11	<b>Port 11 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-273. 668h Port Enable Status  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
13	<b>Port 13 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
14	<b>Port 14 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
15	<b>Port 15 Enable Status</b> 0 = Port is disabled 1 = Port is enabled	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 14-36. Port Configurations**

STRAP_PORTCFG[3:0]	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15	Port Enable Status Register Value (Port 0, and also the NT Port if Port 0 is a Legacy NT Port, Offset 668h)
(default) 0h	x1	x1	x1	x1	x1	x1	Program the upper 16 bits to FFFFh. To determine the value for the lower 16 bits, start with 0h and Set only those bit positions that correspond to the enabled Ports.										
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	x1	
2h	x4	x4	x1	x1							x1	x1	x1	x1	x1	x1	
3h	x4	x4	x4	x1								x1	x1	x1	x1	x1	
4h	x4	x4	x4	x4													
5h	x8	x1		x1		x1		x1									
6h	x8	x4		x1								x1		x1		x1	
7h	x8	x4		x4													
8h	x8	x8															

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-274. 66Ch Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Negotiated Link Width for Port 0</b> 000b = x1 001b = x2 010b = x4 011b = x8 All other encodings are <i>reserved</i> .	RO	No	000b
3	<b>Link Speed for Port 0</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
6:4	<b>Negotiated Link Width for Port 1</b> 000b = x1 010b = x2 010b = x4 011b = x8 All other encodings are <i>reserved</i> .	RO	No	000b
7	<b>Valid Negotiated Link Width for Port 1</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
10:8	<b>Negotiated Link Width for Port 2</b> 000b = x1 010b = x2 010b = x4 All other encodings are <i>reserved</i> .	RO	No	000b
11	<b>Valid Negotiated Link Width for Port 2</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
14:12	<b>Negotiated Link Width for Port 3</b> 000b = x1 001b = x2 010b = x4 All other encodings are <i>reserved</i> .	RO	No	000b
15	<b>Valid Negotiated Link Width for Port 3</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-274. 66Ch Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Negotiated Link Width for Port 4</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
19	<b>Link Speed for Port 4</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
22:20	<b>Negotiated Link Width for Port 5</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
23	<b>Valid Negotiated Link Width for Port 5</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
26:24	<b>Negotiated Link Width for Port 6</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
27	<b>Valid Negotiated Link Width for Port 6</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
30:28	<b>Negotiated Link Width for Port 7</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
31	<b>Valid Negotiated Link Width for Port 7</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-275. 670h Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Negotiated Link Width for Port 8</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
3	<b>Valid Negotiated Link Width for Port 8</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
6:4	<b>Negotiated Link Width for Port 9</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
7	<b>Valid Negotiated Link Width for Port 9</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
10:8	<b>Negotiated Link Width for Port 10</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
11	<b>Valid Negotiated Link Width for Port 10</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
14:12	<b>Negotiated Link Width for Port 11</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
15	<b>Valid Negotiated Link Width for Port 11</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-275. 670h Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15  
(Port 0, and also the NT Port if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Negotiated Link Width for Port 12</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
19	<b>Valid Negotiated Link Width for Port 12</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
22:20	<b>Negotiated Link Width for Port 13</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
23	<b>Valid Negotiated Link Width for Port 13</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
26:24	<b>Negotiated Link Width for Port 14</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
27	<b>Valid Negotiated Link Width for Port 14</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0
30:28	<b>Negotiated Link Width for Port 15</b> 000b = x1 All other encodings are <i>reserved</i> .	RO	No	000b
31	<b>Valid Negotiated Link Width for Port 15</b> 0 = Negotiated Link SerDes speed is 2.5 GT/s 1 = Negotiated Link SerDes speed is 5.0 GT/s	RO	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-276. 674h Port Cut-Thru Enable Status  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
1	<b>Port 1 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
2	<b>Port 2 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
3	<b>Port 3 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-276. 674h Port Cut-Thru Enable Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 4 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
5	<b>Port 5 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
6	<b>Port 6 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
7	<b>Port 7 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-276. 674h Port Cut-Thru Enable Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
9	<b>Port 9 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
10	<b>Port 10 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
11	<b>Port 11 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-276. 674h Port Cut-Thru Enable Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
13	<b>Port 13 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
14	<b>Port 14 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
15	<b>Port 15 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)	RO	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 574h[3:0])
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 14-277. 678h Ingress PLL RAM ECC 1-Bit Counter  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
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.	RO	No	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-278. 67Ch Ingress Control 2  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b><i>Factory Test Only</i></b>	RWS	Yes	0-0h
6	<b>tic_dma_tc_vc_map_en</b> Allows DMA to use VC1 resources in Ingress, based upon the programmed Traffic Class.	RWS	Yes	0
14:7	<b>tic_dma_tc_vc_map</b> Each bit enables the corresponding Traffic Class to use VC1 resources in the Ingress.	RWS	Yes	0
15	<b><i>Factory Test Only</i></b>	RWS	Yes	1
18:16	<b><i>Factory Test Only</i></b>	RWS	Yes	000b
31:19	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 14.16.6 Device-Specific Registers – IOCAM Base and Limit Upper 16 Bits (Offsets 680h – 6BCh)

This section details the Device-Specific IOCAM Base and Limit Upper 16-Bit registers. Table 14-37 defines the register map.

**Table 14-37. Device-Specific IOCAM Base and Limit Upper 16 Bits Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
	IOCAM UP 0	680h
	IOCAM UP 1	684h
	IOCAM UP 2	688h
	IOCAM UP 3	68Ch
	IOCAM UP 4	690h
	IOCAM UP 5	694h
	IOCAM UP 6	698h
	IOCAM UP 7	69Ch
	IOCAM UP 8	6A0h
	IOCAM UP 9	6A4h
	IOCAM UP 10	6A8h
	IOCAM UP 11	6ACh
	IOCAM UP 12	6B0h
	IOCAM UP 13	6B4h
	IOCAM UP 14	6B8h
	IOCAM UP 15	6BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-279. 680h IOCAM UP 0**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-280. 684h IOCAM UP 1**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-281. 688h IOCAM UP 2**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-282. 68Ch IOCAM UP 3**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-283. 690h IOCAM UP 4**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-284. 694h IOCAM UP 5**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-285. 698h IOCAM UP 6**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-286. 69Ch IOCAM UP 7**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-287. 6A0h IOCAM UP 8**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-288. 6A4h IOCAM UP 9**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-289. 6A8h IOCAM UP 10**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-290. 6ACh IOCAM UP 11**

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-291. 6B0h IOCAM UP 12**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-292. 6B4h IOCAM UP 13**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-293. 6B8h IOCAM UP 14**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

**Register 14-294. 6BCh IOCAM UP 15**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>I/O Base Upper 16 Bits</b> I/O Base Upper 16 bits, IOCAM Base[31:16].	RW	Yes	0000h
31:16	<b>I/O Limit Upper 16 Bits</b> I/O Limit Upper 16 bits, IOCAM Limit[31:16].	RW	Yes	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

#### **14.16.7 Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch)**

This section details the Device-Specific Base Address Shadow registers, which are a shadow copy of the two Type 1 Configuration Base Address registers (**BAR0** and **BAR1**) for each PEX 8619 Port. [Table 14-38](#) defines the register map.

**Table 14-38. Device-Specific BAR Shadow Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

**Table 14-38. Device-Specific BAR Shadow Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
		BAR0 Shadow for Port 12
		BAR1 Shadow for Port 12
		BAR0 Shadow for Port 13
		BAR1 Shadow for Port 13
		BAR0 Shadow for Port 14
		BAR1 Shadow for Port 14
		BAR0 Shadow for Port 15
		BAR1 Shadow for Port 15

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-295. 6C0h BAR0 Shadow for Port 0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 0.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 0 Base Address 0.	RW	Yes	0000h

**Register 14-296. 6C4h BAR1 Shadow for Port 0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 0</b> register <i>Memory Map Type</i> field (offset 6C0h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 0 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 0</b> register <i>Memory Map Type</i> field (offset 6C0h[2:1]) is not programmed to 10b.	RsvdP	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-297. 6C8h BAR0 Shadow for Port 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 1. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 1 Base Address 0.	RW	Yes	0000h

**Register 14-298. 6CCh BAR1 Shadow for Port 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 1</b> register <i>Memory Map Type</i> field (offset 6C8h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 1 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 1</b> register <i>Memory Map Type</i> field (offset 6C8h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-299. 6D0h BAR0 Shadow for Port 2**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 2.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 2 Base Address 0.	RW	Yes	0000h

**Register 14-300. 6D4h BAR1 Shadow for Port 2**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 2</b> register <i>Memory Map Type</i> field (offset 6D0h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 2 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 2</b> register <i>Memory Map Type</i> field (offset 6D0h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-301. 6D8h BAR0 Shadow for Port 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 3. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 3 Base Address 0.	RW	Yes	0000h

**Register 14-302. 6DCh BAR1 Shadow for Port 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 3</b> register <i>Memory Map Type</i> field (offset 6D8h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 3 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 3</b> register <i>Memory Map Type</i> field (offset 6D8h[2:1]) is not programmed to 10b.	RsvdP	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-303. 6E0h BAR0 Shadow for Port 4  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 4.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 4 Base Address 0.	RW	Yes	0000h

**Register 14-304. 6E4h BAR1 Shadow for Port 4  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 4</b> register <i>Memory Map Type</i> field (offset 6E0h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 4 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 4</b> register <i>Memory Map Type</i> field (offset 6E0h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-305. 6E8h BAR0 Shadow for Port 5  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 5. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 5 Base Address 0.	RW	Yes	0000h

**Register 14-306. 6ECh BAR1 Shadow for Port 5  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 5</b> register <i>Memory Map Type</i> field (offset 6E8h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 5 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 5</b> register <i>Memory Map Type</i> field (offset 6E8h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-307. 6F0h BAR0 Shadow for Port 6  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 6.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 6 Base Address 0.	RW	Yes	0000h

**Register 14-308. 6F4h BAR1 Shadow for Port 6  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 6</b> register <i>Memory Map Type</i> field (offset 6F0h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 6 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 6</b> register <i>Memory Map Type</i> field (offset 6F0h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-309. 6F8h BAR0 Shadow for Port 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 7. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 7 Base Address 0.	RW	Yes	0000h

**Register 14-310. 6FCh BAR1 Shadow for Port 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 7</b> register <i>Memory Map Type</i> field (offset 6F8h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 7 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 7</b> register <i>Memory Map Type</i> field (offset 6F8h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-311. 700h BAR0 Shadow for Port 8**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 8.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 8 Base Address 0.	RW	Yes	0000h

**Register 14-312. 704h BAR1 Shadow for Port 8**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 8</b> register <i>Memory Map Type</i> field (offset 700h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 8 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 8</b> register <i>Memory Map Type</i> field (offset 700h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-313. 708h BAR0 Shadow for Port 9  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 9. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 9 Base Address 0.	RW	Yes	0000h

**Register 14-314. 70Ch BAR1 Shadow for Port 9  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 9</b> register <i>Memory Map Type</i> field (offset 708h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 9 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 9</b> register <i>Memory Map Type</i> field (offset 708h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-315. 710h BAR0 Shadow for Port 10**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 10.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 10 Base Address 0.	RW	Yes	0000h

**Register 14-316. 714h BAR1 Shadow for Port 10**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 10</b> register <i>Memory Map Type</i> field (offset 710h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 10 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 10</b> register <i>Memory Map Type</i> field (offset 710h[2:1]) is not programmed to 10b.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-317. 718h BAR0 Shadow for Port 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 11. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 11 Base Address 0.	RW	Yes	0000h

**Register 14-318. 71Ch BAR1 Shadow for Port 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 11</b> register <i>Memory Map Type</i> field (offset 718h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 11 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 11</b> register <i>Memory Map Type</i> field (offset 718h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-319. 720h BAR0 Shadow for Port 12**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 12.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 12 Base Address 0.	RW	Yes	0000h

**Register 14-320. 724h BAR1 Shadow for Port 12**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 12</b> register <i>Memory Map Type</i> field (offset 720h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 12 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 12</b> register <i>Memory Map Type</i> field (offset 720h[2:1]) is not programmed to 10b.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-321. 728h BAR0 Shadow for Port 13  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 13. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 13 Base Address 0.	RW	Yes	0000h

**Register 14-322. 72Ch BAR1 Shadow for Port 13  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 13</b> register <i>Memory Map Type</i> field (offset 728h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 13 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 13</b> register <i>Memory Map Type</i> field (offset 728h[2:1]) is not programmed to 10b.	RsvdP	No	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-323. 730h BAR0 Shadow for Port 14**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR  <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 14.  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable  <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 14 Base Address 0.	RW	Yes	0000h

**Register 14-324. 734h BAR1 Shadow for Port 14**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 14</b> register <i>Memory Map Type</i> field (offset 730h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 14 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 14</b> register <i>Memory Map Type</i> field (offset 730h[2:1]) is not programmed to 10b.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-325. 738h BAR0 Shadow for Port 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR <i>Note:</i> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> Memory Mapping for Port 15. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Not Prefetchable 1 = Prefetchable <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	000h
31:17	<b>Base Address 0</b> Shadow copy of Port 15 Base Address 0.	RW	Yes	0000h

**Register 14-326. 73Ch BAR1 Shadow for Port 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1[63:32]</b> When the <b>BAR0 Shadow for Port 15</b> register <i>Memory Map Type</i> field (offset 738h[2:1]) is programmed to 10b, 64-bit mode is enabled and this register becomes a Shadow copy of Port 15 Base Address 1[63:32].	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 Shadow for Port 15</b> register <i>Memory Map Type</i> field (offset 738h[2:1]) is not programmed to 10b.			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.8 Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch)

This section details the Device-Specific Virtual Channel (VC) Resource Control Shadow registers, for each Port (which shadow the VC0 and VC1 registers at offsets 15Ch and 168h, respectively), as well as DMA BAR, and DMA Read and Write Latency registers. [Table 14-39](#) defines the register map.

**Table 14-39. Device-Specific VC Resource Control Shadow, DMA BAR, and DMA Read and Write Latency Register Map (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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 0 VC0 Resource Control
		Port 0 VC1 Resource Control
		Port 1 VC0 Resource Control
		Port 1 VC1 Resource Control
		Port 2 VC0 Resource Control
		Port 2 VC1 Resource Control
		Port 3 VC0 Resource Control
		Port 3 VC1 Resource Control
		Port 4 VC0 Resource Control
		Port 4 VC1 Resource Control
		Port 5 VC0 Resource Control
		Port 5 VC1 Resource Control
		Port 6 VC0 Resource Control
		Port 6 VC1 Resource Control
		Port 7 VC0 Resource Control
		Port 7 VC1 Resource Control
		Port 8 VC0 Resource Control
		Port 8 VC1 Resource Control
		Port 9 VC0 Resource Control
		Port 9 VC1 Resource Control
		Port 10 VC0 Resource Control
		Port 10 VC1 Resource Control
		Port 11 VC0 Resource Control
		Port 11 VC1 Resource Control
		Port 12 VC0 Resource Control
		Port 12 VC1 Resource Control
		Port 13 VC0 Resource Control
		Port 13 VC1 Resource Control

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 14-39. Device-Specific VC Resource Control Shadow, DMA BAR, and DMA Read and Write Latency Register Map (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
		Port 14 VC0 Resource Control 7B0h
		Port 14 VC1 Resource Control 7B4h
		Port 15 VC0 Resource Control 7B8h
		Port 15 VC1 Resource Control 7BCh
		Base Address 0 DMA 7C0h
		Base Address 1 DMA 7C4h
		DMA Minimum and Maximum Read Latency 7C8h
		DMA Minimum and Maximum Write Latency 7CCh
		<i>Reserved</i> 7D0h – 83Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-327. 740h Port 0 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 0</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 for Port 0</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 0</b> Selects the Port Arbitration type for Port 0. 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.  When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0.  0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	001b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 0</b> Defines the Port 0 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 0</b> 0 = Not allowed 1 = Enables the Port 0 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-328. 744h Port 0 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 0</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 0</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 0</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 0</b> Defines the Port 0 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 0</b> 0 = Disables Port 0 VC1 1 = Enables Port 0 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-329. 748h Port 1 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 1</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 1</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 1</b> Selects the Port Arbitration type for Port 1. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 1</b> Defines the Port 1 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 1</b> 0 = Not allowed 1 = Enables the Port 1 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-330. 74Ch Port 1 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 1</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 1</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 1</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 1</b> Defines the Port 1 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 1</b> 0 = Disables Port 1 VC1 1 = Enables Port 1 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-331. 750h Port 2 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 2</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 2</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 2</b> Selects the Port Arbitration type for Port 2. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 2</b> Defines the Port 2 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 2</b> 0 = Not allowed 1 = Enables the Port 2 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-332. 754h Port 2 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 2</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 2</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 2</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 2</b> Defines the Port 2 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 2</b> 0 = Disables Port 2 VC1 1 = Enables Port 2 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-333. 758h Port 3 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 3</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 3</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 3</b> Selects the Port Arbitration type for Port 3. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 3</b> Defines the Port 3 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 3</b> 0 = Not allowed 1 = Enables the Port 3 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-334. 75Ch Port 3 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 3</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 3</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 3</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 3</b> Defines the Port 3 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 3</b> 0 = Disables Port 3 VC1 1 = Enables Port 3 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-335. 760h Port 4 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 4</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 4</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 4</b> Selects the Port Arbitration type for Port 4. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 4</b> Defines the Port 4 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 4</b> 0 = Not allowed 1 = Enables the Port 4 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-336. 764h Port 4 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 4</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 4</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 4</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 4</b> Defines the Port 4 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 4</b> 0 = Disables Port 4 VC1 1 = Enables Port 4 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-337. 768h Port 5 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 5</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 5</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 5</b> Selects the Port Arbitration type for Port 5. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 5</b> Defines the Port 5 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 5</b> 0 = Not allowed 1 = Enables the Port 5 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-338. 76Ch Port 5 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 5</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 5</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 5</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 5</b> Defines the Port 5 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 5</b> 0 = Disables Port 5 VC1 1 = Enables Port 5 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-339. 770h Port 6 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 6</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 6</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 6</b> Selects the Port Arbitration type for Port 6. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 6</b> Defines the Port 6 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 6</b> 0 = Not allowed 1 = Enables the Port 6 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-340. 774h Port 6 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 6</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 6</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 6</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 6</b> Defines the Port 6 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 6</b> 0 = Disables Port 6 VC1 1 = Enables Port 6 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-341. 778h Port 7 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 7</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 7</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 7</b> Selects the Port Arbitration type for Port 7. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 7</b> Defines the Port 7 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 7</b> 0 = Not allowed 1 = Enables the Port 7 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-342. 77Ch Port 7 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 7</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 7</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 7</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 7</b> Defines the Port 7 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 7</b> 0 = Disables Port 7 VC1 1 = Enables Port 7 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-343. 780h Port 8 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 8</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 8</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 8</b> Selects the Port Arbitration type for Port 8. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 8</b> Defines the Port 8 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 8</b> 0 = Not allowed 1 = Enables the Port 8 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-344. 784h Port 8 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 8</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 8</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 8</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 8</b> Defines the Port 8 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 8</b> 0 = Disables Port 8 VC1 1 = Enables Port 8 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-345. 788h Port 9 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 9</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 9</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 9</b> Selects the Port Arbitration type for Port 9. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 9</b> Defines the Port 9 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 9</b> 0 = Not allowed 1 = Enables the Port 9 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-346. 78Ch Port 9 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 9</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 9</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 9</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 9</b> Defines the Port 9 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 9</b> 0 = Disables Port 9 VC1 1 = Enables Port 9 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-347. 790h Port 10 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 10</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 10</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 10</b> Selects the Port Arbitration type for Port 10. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 10</b> Defines the Port 10 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 10</b> 0 = Not allowed 1 = Enables the Port 10 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-348. 794h Port 10 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 10</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 10</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 10</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 10</b> Defines the Port 10 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 10</b> 0 = Disables Port 10 VC1 1 = Enables Port 10 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-349. 798h Port 11 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 11</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 for Port 11</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 11</b> Selects the Port Arbitration type for Port 11. 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. When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	001b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 11</b> Defines the Port 11 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 11</b> 0 = Not allowed 1 = Enables the Port 11 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-350. 79Ch Port 11 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 11</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 11</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 11</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 11</b> Defines the Port 11 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 11</b> 0 = Disables Port 11 VC1 1 = Enables Port 11 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-351. 7A0h Port 12 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 12</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 12</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 12</b> Selects the Port Arbitration type for Port 12. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 12</b> Defines the Port 12 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 12</b> 0 = Not allowed 1 = Enables the Port 12 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-352. 7A4h Port 12 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 12</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 12</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 12</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 12</b> Defines the Port 12 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 12</b> 0 = Disables Port 12 VC1 1 = Enables Port 12 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-353. 7A8h Port 13 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 13</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 13</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 13</b> Selects the Port Arbitration type for Port 13. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 13</b> Defines the Port 13 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 13</b> 0 = Not allowed 1 = Enables the Port 13 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-354. 7ACh Port 13 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 13</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 13</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 13</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 13</b> Defines the Port 13 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 13</b> 0 = Disables Port 13 VC1 1 = Enables Port 13 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-355. 7B0h Port 14 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 14</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 14</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 14</b> Selects the Port Arbitration type for Port 14. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 14</b> Defines the Port 14 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 14</b> 0 = Not allowed 1 = Enables the Port 14 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-356. 7B4h Port 14 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 14</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 14</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 14</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 14</b> Defines the Port 14 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 14</b> 0 = Disables Port 14 VC1 1 = Enables Port 14 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-357. 7B8h Port 15 VC0 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC0 Map for Port 15</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0.	RO	No	1
7:1	Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 15</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns 0.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 15</b> Selects the Port Arbitration type for Port 15. 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.	RW	Yes	001b
	When WRR is enabled, the Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3; if the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. 0 = Round-Robin (Hardware-Fixed) arbitration scheme 1 = WRR with 32 Phases	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC0 ID for Port 15</b> Defines the Port 15 VC0 ID code. This field is Cleared, because VC0 is the default VC.	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC0 Enable for Port 15</b> 0 = Not allowed 1 = Enables the Port 15 default VC, VC0	RO	No	1

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-358. 7BCh Port 15 VC1 Resource Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC1 Map for Port 15</b> Defines Traffic Classes [7:1], respectively, and indicates which TCs are mapped into VC1. Traffic Class 0 must be mapped to VC0. Traffic Classes [7:1] can be mapped to VC1.	RO	No	00h
7:1		RW	Yes	
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table for Port 15</b> <i>Not supported</i> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select for Port 15</b> <i>Not supported</i>	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC1 ID for Port 15</b> Defines the Port 15 VC1 ID code.	RW	Yes	001b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC1 Enable for Port 15</b> 0 = Disables Port 15 VC1 1 = Enables Port 15 VC1	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-359. 7C0h Base Address 0 DMA**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Base Address register maps the PEX 8619 Configuration registers into Memory space <i>Note:</i> Hardwired to 0.	RO	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 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8619 Configuration registers into Non-Prefetchable Memory space <i>Note:</i> Hardwired to 0.	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	0-0h
31:17	<b>Base Address 0 DMA</b> Base Address ( <b>BAR0 DMA</b> ) for Device-Specific Memory-Mapped Configuration mechanism.	RW	Yes	0000h

**Register 14-360. 7C4h Base Address 1 DMA**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1 DMA</b> <i>Reserved</i> when the <b>Base Address 0 DMA</b> register is not enabled as a 64-bit BAR ( <i>Memory Map Type</i> field (offset 7C0h[2:1]) is not programmed to 10b).	RsvdP	No	0000_0000h
	For 64-bit addressing, Base Address 1 DMA ( <b>BAR1 DMA</b> ) extends Base Address 0 DMA to provide the upper 32 Address bits when the <b>Base Address 0 DMA</b> register <i>Memory Map Type</i> field (offset 7C0h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-361. 7C8h DMA Minimum and Maximum Read Latency  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Measures Read Latency. Counter starts counting when a Read is issued, and stops counting when the last Completion is received.				
15:0	<b>DMA Minimum Read Latency</b> DMA Read Latency Counter. Updates when the last phase of a Completion is received for a given Read Request. If the current Completion Latency is less than the register value, the current Completion Latency becomes the new Minimum Read Latency. The Counter saturates at FFFFh and does not roll over to 0000h.	RW1C	No	FFFFh
31:16	<b>DMA Maximum Read Latency</b> DMA Read Latency Counter. Updates when the last phase of a Completion is received for a given Read Request. If the current Completion Latency is greater than the register value, the current Completion Latency becomes the new Maximum Read Latency. The Counter saturates at FFFFh and does not roll over to 0000h.	RW1C	No	0000h

**Register 14-362. 7CCh DMA Minimum and Maximum Write Latency  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Measures Write Latency. Counter starts counting when a Write is issued, and stops counting when an ACK is received.				
15:0	<b>DMA Minimum Write Latency</b> DMA Write Latency Counter. Updates when a de-allocation for DMA Writes is received. If the current Write Latency is less than the register value, the current Write Latency becomes the new Minimum Write Latency. The Counter saturates at FFFFh and does not roll over to 0000h.	RW1C	No	FFFFh
31:16	<b>DMA Maximum Write Latency</b> DMA Write Latency Counter. Updates when a de-allocation for DMA Writes is received. If the current Write Latency is greater than the register value, the current Write Latency becomes the new Maximum Write Latency. The Counter saturates at FFFFh and does not roll over to 0000h.	RW1C	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.9 Device-Specific Registers – Ingress Credit Handler Port Pool (Offsets 940h – 94Ch)

This section details the Device-Specific Ingress Credit Handler (INCH) Port Pool registers. [Table 14-40](#) defines the register map.

The original intent for the INCH Port Pool registers was to provide another level of reservation of the Common Pool credits. These registers are essentially redundant to what is accomplished by changing the values of the **INCH Threshold** registers. (Refer to [Section 14.16.13](#).)

**Consider the INCH Port Pool registers to be *reserved* and only change the credit Settings, using the INCH Threshold registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.**

Refer to [Section 9.3, “Ingress Resources,”](#) and its subsections, for further details regarding credits and credit allocation.

**Table 14-40. Device-Specific INCH Port Pool Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
	INCH Port Pool for Ports 0, 1, 2, 3	940h
	INCH Port Pool for Ports 4, 5, 6, 7	944h
	INCH Port Pool for Ports 8, 9, 10, 11	948h
	INCH Port Pool for Ports 12, 13, 14, 15	94Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-363. 940h INCH Port Pool for Ports 0, 1, 2, 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Consider the INCH Port Pool registers to be reserved and only change the credit Settings, using the Ingress Credit Handler (INCH) registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.</i>				
2:0	<b>Port 0 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 0. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
3	<b>Unused 0</b> Keep value at 0. Additional bit for Port 0 Payload Pool.	RWS	Yes	0
6:4	<b>Port 0 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 0. 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 Port 0 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-363. 940h INCH Port Pool for Ports 0, 1, 2, 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
10:8	<b>Port 1 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 1. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
11	<b>Unused 2</b> Keep value at 0. Additional bit for Port 1 Payload Pool.	RWS	Yes	0
14:12	<b>Port 1 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 1. 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
15	<b>Unused 3</b> Keep value at 0. Additional bit for Port 1 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-363. 940h INCH Port Pool for Ports 0, 1, 2, 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Port 2 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 2. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
19	<b>Unused 4</b> Keep value at 0. Additional bit for Port 2 Payload Pool.	RWS	Yes	0
22:20	<b>Port 2 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 2. 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
23	<b>Unused 5</b> Keep value at 0. Additional bit for Port 2 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-363. 940h INCH Port Pool for Ports 0, 1, 2, 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 3 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 3. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
27	<b>Unused 6</b> Keep value at 0. Additional bit for Port 3 Payload Pool.	RWS	Yes	0
30:28	<b>Port 3 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 3. 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
31	<b>Unused 7</b> Keep value at 0. Additional bit for Port 3 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-364. 944h INCH Port Pool for Ports 4, 5, 6, 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Consider the INCH Port Pool registers to be reserved and only change the credit Settings, using the Ingress Credit Handler (INCH) registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.</i>				
2:0	<b>Port 4 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 4. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
3	<b>Unused 0</b> Keep value at 0. Additional bit for Port 4 Payload Pool.	RWS	Yes	0
6:4	<b>Port 4 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 4. 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 Port 4 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-364. 944h INCH Port Pool for Ports 4, 5, 6, 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
10:8	<b>Port 5 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 5. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
11	<b>Unused 2</b> Keep value at 0. Additional bit for Port 5 Payload Pool.	RWS	Yes	0
14:12	<b>Port 5 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 5. 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
15	<b>Unused 3</b> Keep value at 0. Additional bit for Port 5 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-364. 944h INCH Port Pool for Ports 4, 5, 6, 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Port 6 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 6. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
19	<b>Unused 4</b> Keep value at 0. Additional bit for Port 6 Payload Pool.	RWS	Yes	0
22:20	<b>Port 6 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 6. 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
23	<b>Unused 5</b> Keep value at 0. Additional bit for Port 6 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-364. 944h INCH Port Pool for Ports 4, 5, 6, 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 7 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 7. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
27	<b>Unused 6</b> Keep value at 0. Additional bit for Port 7 Payload Pool.	RWS	Yes	0
30:28	<b>Port 7 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 7. 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
31	<b>Unused 7</b> Keep value at 0. Additional bit for Port 7 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-365. 948h INCH Port Pool for Ports 8, 9, 10, 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Consider the INCH Port Pool registers to be reserved and only change the credit Settings, using the Ingress Credit Handler (INCH) registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.</i>				
2:0	<b>Port 8 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 8. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
3	<b>Unused 0</b> Keep value at 0. Additional bit for Port 8 Payload Pool.	RWS	Yes	0
6:4	<b>Port 8 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 8. 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 Port 8 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-365. 948h INCH Port Pool for Ports 8, 9, 10, 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
10:8	<b>Port 9 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 9. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
11	<b>Unused 2</b> Keep value at 0. Additional bit for Port 9 Payload Pool.	RWS	Yes	0
14:12	<b>Port 9 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 9. 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
15	<b>Unused 3</b> Keep value at 0. Additional bit for Port 9 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-365. 948h INCH Port Pool for Ports 8, 9, 10, 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Port 10 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 10. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
19	<b>Unused 4</b> Keep value at 0. Additional bit for Port 10 Payload Pool.	RWS	Yes	0
22:20	<b>Port 10 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 10. 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
23	<b>Unused 5</b> Keep value at 0. Additional bit for Port 10 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-365. 948h INCH Port Pool for Ports 8, 9, 10, 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 11 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 11. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
27	<b>Unused 6</b> Keep value at 0. Additional bit for Port 11 Payload Pool.	RWS	Yes	0
30:28	<b>Port 11 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 11. 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
31	<b>Unused 7</b> Keep value at 0. Additional bit for Port 11 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-366. 94Ch INCH Port Pool for Ports 12, 13, 14, 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Consider the INCH Port Pool registers to be reserved and only change the credit Settings, using the Ingress Credit Handler (INCH) registers. Do not change the INCH Port Pool registers from their default values, unless directed otherwise by PLX Technical Support.</i>				
2:0	<b>Port 12 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 12. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
3	<b>Unused 0</b> Keep value at 0. Additional bit for Port 12 Payload Pool.	RWS	Yes	0
6:4	<b>Port 12 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 12. 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 Port 12 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-366. 94Ch INCH Port Pool for Ports 12, 13, 14, 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
10:8	<b>Port 13 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 13. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
11	<b>Unused 2</b> Keep value at 0. Additional bit for Port 13 Payload Pool.	RWS	Yes	0
14:12	<b>Port 13 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 13. 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
15	<b>Unused 3</b> Keep value at 0. Additional bit for Port 13 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-366. 94Ch INCH Port Pool for Ports 12, 13, 14, 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>Port 14 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 14. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
19	<b>Unused 4</b> Keep value at 0. Additional bit for Port 14 Payload Pool.	RWS	Yes	0
22:20	<b>Port 14 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 14. 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
23	<b>Unused 5</b> Keep value at 0. Additional bit for Port 14 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-366. 94Ch INCH Port Pool for Ports 12, 13, 14, 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 15 Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs dedicated to Port 15. 000b = 0 001b = 8 010b = 16 011b = 24 100b = 32 101b = 64 110b, 111b = 128	RWS	Yes	000b
27	<b>Unused 6</b> Keep value at 0. Additional bit for Port 15 Payload Pool.	RWS	Yes	0
30:28	<b>Port 15 Header Pool</b> Combined Header credits (other than the initial credits) dedicated to Port 15. 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
31	<b>Unused 7</b> Keep value at 0. Additional bit for Port 15 Header Pool.	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 14.16.10 Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)

This section details the Device-Specific, Vendor-Specific Extended Capability 2 registers. Table 14-41 defines the register map.

**Table 14-41. Device-Specific, Vendor-Specific Extended Capability 2 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 2 (000h)	Capability Version 2 (1h)	PCI Express Extended Capability ID 2 (000Bh)	950h							
Vendor-Specific Header 2										954h
Hardwired Device ID										958h
<i>Reserved</i>										95Ch
Hardwired Revision ID										

**Register 14-367. 950h 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, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	RO	Yes	000Bh
19:16	<b>Capability Version 2</b>	RO	Yes	1h
31:20	<b>Next Capability Offset 2</b> 000h = This extended capability is the last capability in the PEX 8619 Extended Capabilities list	RO	Yes	000h

**Register 14-368. 954h 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.	RO	Yes	0001h
19:16	<b>Vendor-Specific Rev 2</b> Version Number of this structure.	RO	Yes	0h
31:20	<b>Vendor-Specific Length 2</b> Quantity of bytes in the entire structure.	RO	Yes	010h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-369. 958h 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 <a href="#">Vendor ID</a> value, 10B5h.	RO	No	10B5h
31:16	<b>Hardwired Device ID</b> Always returns the PEX 8619 default <a href="#">Device ID</a> value, 8619h.	RO	No	8619h

**Register 14-370. 95Ch 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 8619 default <a href="#">PCI Revision ID</a> value, BAh.	RO	No	Current Rev # (BAh)
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 14.16.11 Device-Specific Registers – DMA Port Credit (Offsets 978h – 97Ch)

This section details the Device-Specific DMA Port Credit registers. [Table 14-42](#) defines the register map.

**Table 14-42. Device-Specific DMA Port Credit Register Map  
(DMA of Upstream 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	
	Even Ports – DMA Port Credit	978h
	Odd Ports – DMA Port Credit	97Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-371. 978h Even Ports – DMA Port Credit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
DMA credits are used for DMA Memory Write and Message transactions. This register applies to the even-numbered Ports.				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<b>DMA Payload Credit</b> Default advertised Payload credit for DMA. Actual value is dependent upon Link width and Port configuration.	RWS	Yes	0 (DMA is disabled) 64 (DMA is enabled)
15:9	<b>DMA Header Credit</b> Default advertised Header credit for DMA. Actual value is dependent upon Link width and Port configuration.	RWS	Yes	0 (DMA is disabled) 4 (DMA is enabled)
17:16	<b>UpdateFC High-Priority Threshold for DMA Payload Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for DMA Header Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
21:20	<b>DMA Congested Port Weight</b> Refer to <a href="#">Section 9.3.5, “Credit Allocation When Common Pool Is Consumed,”</a> for details.  00b = x1 01b = x2 10b = x4 11b = x8	RWS	Yes	00b
22	<b>DMA Congested Port Weight Enable</b> 0 = Disabled 1 = Enabled	RWS	Yes	0
31:23	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-372. 97Ch Odd Ports – DMA Port Credit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
DMA credits are used for DMA Memory Write and Message transactions. This register applies to the odd-numbered Ports.				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<b>DMA Payload Credit</b> Default advertised Payload credit for DMA. Actual value is dependent upon Link width and Port configuration.	RWS	Yes	0 (DMA is disabled) 64 (DMA is enabled)
15:9	<b>DMA Header Credit</b> Default advertised Header credit for DMA. Actual value is dependent upon Link width and Port configuration.	RWS	Yes	0 (DMA is disabled) 4 (DMA is enabled)
17:16	<b>UpdateFC High-Priority Threshold for DMA Payload Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for DMA Header Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
21:20	<b>DMA Congested Port Weight</b> Refer to <a href="#">Section 9.3.5, “Credit Allocation When Common Pool Is Consumed,”</a> for details.  00b = x1 01b = x2 10b = x4 11b = x8	RWS	Yes	00b
22	<b>DMA Congested Port Weight Enable</b> 0 = Disabled 1 = Enabled	RWS	Yes	0
31:23	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.12 Device-Specific Registers – ACS Extended Capability (Offsets 980h – 9FCh)

This section details the Device-Specific ACS Extended Capability registers. The registers in this structure are shadow copies of the Port 0 registers located at offsets 524h and 528h, for each Port. Table 14-43 defines the register map.

**Table 14-43. Device-Specific ACS Extended Capability Register Map  
(Port 0)**

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	
ACS Port 0 Control	ACS Port 0 Capability	980h
<i>Reserved</i>	ACS Port 0 Egress Control Vector	984h
ACS Port 1 Control	ACS Port 1 Capability	988h
<i>Reserved</i>	ACS Port 1 Egress Control Vector	98Ch
ACS Port 2 Control	ACS Port 2 Capability	990h
<i>Reserved</i>	ACS Port 2 Egress Control Vector	994h
ACS Port 3 Control	ACS Port 3 Capability	998h
<i>Reserved</i>	ACS Port 3 Egress Control Vector	99Ch
ACS Port 4 Control	ACS Port 4 Capability	9A0h
<i>Reserved</i>	ACS Port 4 Egress Control Vector	9A4h
ACS Port 5 Control	ACS Port 5 Capability	9A8h
<i>Reserved</i>	ACS Port 5 Egress Control Vector	9ACh
ACS Port 6 Control	ACS Port 6 Capability	9B0h
<i>Reserved</i>	ACS Port 6 Egress Control Vector	9B4h
ACS Port 7 Control	ACS Port 7 Capability	9B8h
<i>Reserved</i>	ACS Port 7 Egress Control Vector	9BCh
ACS Port 8 Control	ACS Port 8 Capability	9C0h
<i>Reserved</i>	ACS Port 8 Egress Control Vector	9C4h
ACS Port 9 Control	ACS Port 9 Capability	9C8h
<i>Reserved</i>	ACS Port 9 Egress Control Vector	9CCh
ACS Port 10 Control	ACS Port 10 Capability	9D0h
<i>Reserved</i>	ACS Port 10 Egress Control Vector	9D4h
ACS Port 11 Control	ACS Port 11 Capability	9D8h
<i>Reserved</i>	ACS Port 11 Egress Control Vector	9DCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 14-43. Device-Specific ACS Extended Capability Register Map (Port 0) (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	
ACS Port 12 Control	ACS Port 12 Capability	9E0h
<i>Reserved</i>	ACS Port 12 Egress Control Vector	9E4h
ACS Port 13 Control	ACS Port 13 Capability	9E8h
<i>Reserved</i>	ACS Port 13 Egress Control Vector	9ECh
ACS Port 14 Control	ACS Port 14 Capability	9F0h
<i>Reserved</i>	ACS Port 14 Egress Control Vector	9F4h
ACS Port 15 Control	ACS Port 15 Capability	9F8h
<i>Reserved</i>	ACS Port 15 Egress Control Vector	9FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-373. 980h ACS Port 0 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 0 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-373. 980h ACS Port 0 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 0 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 <i>(ACS Source Validation)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 <i>(ACS Translation Blocking)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 <i>(ACS P2P Request Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 <i>(ACS P2P Completion Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-373. 980h ACS Port 0 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-374. 984h ACS Port 0 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 0 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 980h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-374. 984h ACS Port 0 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-375. 988h ACS Port 1 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 1 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-375. 988h ACS Port 1 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 1 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-375. 988h ACS Port 1 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-376. 98Ch ACS Port 1 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 1 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 988h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-376. 98Ch ACS Port 1 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-377. 990h ACS Port 2 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 2 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-377. 990h ACS Port 2 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 2 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 <i>(ACS Source Validation)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 <i>(ACS Translation Blocking)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 <i>(ACS P2P Request Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 <i>(ACS P2P Completion Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-377. 990h ACS Port 2 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-378. 994h ACS Port 2 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 2 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 990h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-378. 994h ACS Port 2 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-379. 998h ACS Port 3 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 3 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-379. 998h ACS Port 3 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 3 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-379. 998h ACS Port 3 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-380. 99Ch ACS Port 3 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 3 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 998h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-380. 99Ch ACS Port 3 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-381. 9A0h ACS Port 4 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 4 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-381. 9A0h ACS Port 4 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 4 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-381. 9A0h ACS Port 4 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-382. 9A4h ACS Port 4 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 4 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <a href="#">9A0h[21]</a> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-382. 9A4h ACS Port 4 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-383. 9A8h ACS Port 5 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 5 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-383. 9A8h ACS Port 5 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 5 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-383. 9A8h ACS Port 5 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-384. 9ACh ACS Port 5 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 5 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 9A8h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-384. 9ACh ACS Port 5 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-385. 9B0h ACS Port 6 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 6 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-385. 9B0h ACS Port 6 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 6 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-385. 9B0h ACS Port 6 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-386. 9B4h ACS Port 6 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 6 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <a href="#">9B0h[21]</a> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-386. 9B4h ACS Port 6 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-387. 9B8h ACS Port 7 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 7 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-387. 9B8h ACS Port 7 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 7 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-387. 9B8h ACS Port 7 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-388. 9BCh ACS Port 7 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 7 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 9B8h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-388. 9BCh ACS Port 7 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-389. 9C0h ACS Port 8 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 8 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-389. 9C0h ACS Port 8 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 8 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-389. 9C0h ACS Port 8 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-390. 9C4h ACS Port 8 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 8 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <a href="#">9C0h[21]</a> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-390. 9C4h ACS Port 8 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-391. 9C8h ACS Port 9 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 9 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-391. 9C8h ACS Port 9 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 9 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-391. 9C8h ACS Port 9 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-392. 9CCh ACS Port 9 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 9 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 9C8h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-392. 9CCCh ACS Port 9 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-393. 9D0h ACS Port 10 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 10 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-393. 9D0h ACS Port 10 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 10 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-393. 9D0h ACS Port 10 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-394. 9D4h ACS Port 10 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 10 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <b>9D0h[21]</b> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-394. 9D4h ACS Port 10 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-395. 9D8h ACS Port 11 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 11 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-395. 9D8h ACS Port 11 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 11 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-395. 9D8h ACS Port 11 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-396. 9DCh ACS Port 11 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 11 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 9D8h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-396. 9DCh ACS Port 11 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-397. 9E0h ACS Port 12 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 12 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-397. 9E0h ACS Port 12 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 12 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-397. 9E0h ACS Port 12 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-398. 9E4h ACS Port 12 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 12 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <a href="#">9E0h[21]</a> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-398. 9E4h ACS Port 12 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-399. 9E8h ACS Port 13 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 13 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-399. 9E8h ACS Port 13 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 13 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-399. 9E8h ACS Port 13 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-400. 9ECh ACS Port 13 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 13 Control</b> register <b>ACS Egress Control Enable</b> bit (offset 9E8h[21]) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-400. 9ECh ACS Port 13 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-401. 9F0h ACS Port 14 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 14 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-401. 9F0h ACS Port 14 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 14 Control</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> RW operation is masked by bit 0 ( <i>ACS Source Validation</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 ( <i>ACS Translation Blocking</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 ( <i>ACS P2P Request Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 ( <i>ACS P2P Completion Redirect</i> ).  0 = Disables 1 = Enables	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-401. 9F0h ACS Port 14 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b> <b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
21	<b>Reserved</b> <b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
22	<b>Reserved</b> <b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-402. 9F4h ACS Port 14 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 14 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <b>9F0h[21]</b> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-402. 9F4h ACS Port 14 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-403. 9F8h ACS Port 15 Control and Capability  
(Port 0)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 15 Capability</b>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress Control.	Downstream	RO	Yes	1
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<b>Reserved</b>		RsvdP	No	0
12:8	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Port configuration-dependent. Sum of all Ports. Maximum value is Ch. <i>Note:</i> The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.	Downstream	RO	Yes	10h
15:13	<b>Reserved</b>		RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-403. 9F8h ACS Port 15 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Port 15 Control</b>					
16	<b>Reserved</b> <b>ACS Source Validation Enable</b> RW operation is masked by bit 0 <i>(ACS Source Validation)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
17	<b>Reserved</b> <b>ACS Translation Blocking Enable</b> RW operation is masked by bit 1 <i>(ACS Translation Blocking)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
18	<b>Reserved</b> <b>ACS P2P Request Redirect Enable</b> RW operation is masked by bit 2 <i>(ACS P2P Request Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0
19	<b>Reserved</b> <b>ACS P2P Completion Redirect Enable</b> RW operation is masked by bit 3 <i>(ACS P2P Completion Redirect)</i> . 0 = Disables 1 = Enables	Upstream  Downstream	RsvdP  RW	No  Yes	0  0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-403. 9F8h ACS Port 15 Control and Capability  
(Port 0) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> RW operation is masked by bit 4 ( <i>ACS Upstream Forwarding</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Egress Control Enable</b> RW operation is masked by bit 5 ( <i>ACS P2P Egress Control</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> RW operation is masked by bit 6 ( <i>ACS Direct Translated P2P</i> ). 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-404. 9FCh ACS Port 15 Egress Control Vector  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The Peer-to-Peer Port x Control bits are valid when the <b>ACS Port 15 Control</b> register <b>ACS Egress Control Enable</b> bit (offset <b>9F8h[21]</b> ) is Set.				
0	<b>Port 0 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
1	<b>Port 1 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
2	<b>Port 2 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
3	<b>Port 3 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
4	<b>Port 4 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
5	<b>Port 5 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
6	<b>Port 6 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
7	<b>Port 7 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
8	<b>Port 8 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
9	<b>Port 9 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
10	<b>Port 10 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
11	<b>Port 11 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-404. 9FCh ACS Port 15 Egress Control Vector  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 12 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
13	<b>Port 13 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
14	<b>Port 14 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
15	<b>Port 15 ACS Egress Control Vector</b> 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	RW	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.13 Device-Specific Registers – Ingress Credit Handler Threshold (Offsets A00h – B7Ch)

This section details the Device-Specific Ingress Credit Handler (INCH) Threshold registers. **Changing credit values from default register values must be done carefully; otherwise the PEX 8619 will not properly function.** Refer to [Section 9.3, “Ingress Resources,”](#) and its subsections, for detailed information regarding rules associated with changing the **INCH Threshold** registers. [Table 14-44](#) defines the register map.

**Table 14-44. Device-Specific INCH Threshold Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
		INCH Threshold Port 0 VC0 Posted
		INCH Threshold Port 0 VC0 Non-Posted
		INCH Threshold Port 0 VC0 Completion
		INCH Threshold Port 0 VC1 Posted
		INCH Threshold Port 0 VC1 Non-Posted
		INCH Threshold Port 0 VC1 Completion
		INCH Threshold Port 1 VC0 Posted
		INCH Threshold Port 1 VC0 Non-Posted
		INCH Threshold Port 1 VC0 Completion
		INCH Threshold Port 1 VC1 Posted
		INCH Threshold Port 1 VC1 Non-Posted
		INCH Threshold Port 1 VC1 Completion
		INCH Threshold Port 2 VC0 Posted
		INCH Threshold Port 2 VC0 Non-Posted
		INCH Threshold Port 2 VC0 Completion
		INCH Threshold Port 2 VC1 Posted
		INCH Threshold Port 2 VC1 Non-Posted
		INCH Threshold Port 2 VC1 Completion
		INCH Threshold Port 3 VC0 Posted
		INCH Threshold Port 3 VC0 Non-Posted
		INCH Threshold Port 3 VC0 Completion
		INCH Threshold Port 3 VC1 Posted
		INCH Threshold Port 3 VC1 Non-Posted
		INCH Threshold Port 3 VC1 Completion
		INCH Threshold Port 4 VC0 Posted
		INCH Threshold Port 4 VC0 Non-Posted
		INCH Threshold Port 4 VC0 Completion
		INCH Threshold Port 4 VC1 Posted
		INCH Threshold Port 4 VC1 Non-Posted
		INCH Threshold Port 4 VC1 Completion
		INCH Threshold Port 5 VC0 Posted
		INCH Threshold Port 5 VC0 Non-Posted

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 14-44. Device-Specific INCH Threshold Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
		INCH Threshold Port 5 VC0 Completion
		INCH Threshold Port 5 VC1 Posted
		INCH Threshold Port 5 VC1 Non-Posted
		INCH Threshold Port 5 VC1 Completion
		INCH Threshold Port 6 VC0 Posted
		INCH Threshold Port 6 VC0 Non-Posted
		INCH Threshold Port 6 VC0 Completion
		INCH Threshold Port 6 VC1 Posted
		INCH Threshold Port 6 VC1 Non-Posted
		INCH Threshold Port 6 VC1 Completion
		INCH Threshold Port 7 VC0 Posted
		INCH Threshold Port 7 VC0 Non-Posted
		INCH Threshold Port 7 VC0 Completion
		INCH Threshold Port 7 VC1 Posted
		INCH Threshold Port 7 VC1 Non-Posted
		INCH Threshold Port 7 VC1 Completion
		INCH Threshold Port 8 VC0 Posted
		INCH Threshold Port 8 VC0 Non-Posted
		INCH Threshold Port 8 VC0 Completion
		INCH Threshold Port 8 VC1 Posted
		INCH Threshold Port 8 VC1 Non-Posted
		INCH Threshold Port 8 VC1 Completion
		INCH Threshold Port 9 VC0 Posted
		INCH Threshold Port 9 VC0 Non-Posted
		INCH Threshold Port 9 VC0 Completion
		INCH Threshold Port 9 VC1 Posted
		INCH Threshold Port 9 VC1 Non-Posted
		INCH Threshold Port 9 VC1 Completion
		INCH Threshold Port 10 VC0 Posted
		INCH Threshold Port 10 VC0 Non-Posted
		INCH Threshold Port 10 VC0 Completion
		INCH Threshold Port 10 VC1 Posted
		INCH Threshold Port 10 VC1 Non-Posted
		INCH Threshold Port 10 VC1 Completion
		INCH Threshold Port 11 VC0 Posted
		INCH Threshold Port 11 VC0 Non-Posted
		INCH Threshold Port 11 VC0 Completion
		INCH Threshold Port 11 VC1 Posted
		INCH Threshold Port 11 VC1 Non-Posted
		INCH Threshold Port 11 VC1 Completion

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 14-44. Device-Specific INCH Threshold Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (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	
	INCH Threshold Port 12 VC0 Posted	B20h
	INCH Threshold Port 12 VC0 Non-Posted	B24h
	INCH Threshold Port 12 VC0 Completion	B28h
	INCH Threshold Port 12 VC1 Posted	B2Ch
	INCH Threshold Port 12 VC1 Non-Posted	B30h
	INCH Threshold Port 12 VC1 Completion	B34h
	INCH Threshold Port 13 VC0 Posted	B38h
	INCH Threshold Port 13 VC0 Non-Posted	B3Ch
	INCH Threshold Port 13 VC0 Completion	B40h
	INCH Threshold Port 13 VC1 Posted	B44h
	INCH Threshold Port 13 VC1 Non-Posted	B48h
	INCH Threshold Port 13 VC1 Completion	B4Ch
	INCH Threshold Port 14 VC0 Posted	B50h
	INCH Threshold Port 14 VC0 Non-Posted	B54h
	INCH Threshold Port 14 VC0 Completion	B58h
	INCH Threshold Port 14 VC1 Posted	B5Ch
	INCH Threshold Port 14 VC1 Non-Posted	B60h
	INCH Threshold Port 14 VC1 Completion	B64h
	INCH Threshold Port 15 VC0 Posted	B68h
	INCH Threshold Port 15 VC0 Non-Posted	B6Ch
	INCH Threshold Port 15 VC0 Completion	B70h
	INCH Threshold Port 15 VC1 Posted	B74h
	INCH Threshold Port 15 VC1 Non-Posted	B78h
	INCH Threshold Port 15 VC1 Completion	B7Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-405. A00h, A18h, A30h, A48h, A60h, A78h, A90h, AA8h, AC0h, AD8h, AF0h, B08h, B20h, B38h, B50h, B68h INCH Threshold Port x VC0 Posted (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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. Port x is Ports 0 through 15 for register offsets A00h, A18h, A30h, A48h, A60h, A78h, A90h, AA8h, AC0h, AD8h, AF0h, B08h, B20h, B38h, B50h, and B68h, respectively. (Refer to <a href="#">Table 14-44</a> .)				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<p><b>Posted Payload Credit</b>            Default advertised Posted Payload credit. Actual value is dependent upon Link width and Port configuration.            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.</p> <p>x1 Port = 64 Payload Credits            x4 Port = 256 Payload Credits            x8 Port = 384 Payload Credits</p>	RWS	Yes	Refer to Description
15:9	<p><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.</p> <p>x1 Port = 7 Header Credits            x4 Port = 26 Header Credits            x8 Port = 49 Header Credits</p>	RWS	Yes	Refer to Description

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-405. A00h, A18h, A30h, A48h, A60h, A78h, A90h, AA8h, AC0h, AD8h, AF0h, B08h, B20h, B38h, B50h, B68h INCH Threshold Port x VC0 Posted  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for Posted Header Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  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. Refer to <a href="#">Section 9.3.5, “Credit Allocation When Common Pool Is Consumed,”</a> for further details.  000b = Request is weighted, based upon the Port’s Link width relative to the effective Link widths of the other Half Station’s 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-406. A04h, A1Ch, A34h, A4Ch, A64h, A7Ch, A94h, AACh, AC4h, ADCh, AF4h, B0Ch, B24h, B3Ch, B54h, B6Ch INCH Threshold Port x VC0 Non-Posted (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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. Port x is Ports 0 through 15 for register offsets A04h, A1Ch, A34h, A4Ch, A64h, A7Ch, A94h, AACh, AC4h, ADCh, AF4h, B0Ch, B24h, B3Ch, B54h, and B6Ch, respectively. (Refer to <a href="#">Table 14-44</a> .)				
8:0	<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. Because of this, the PEX 8619 hardwires this field to 000h (infinite credits).	RsvdP	Yes	000h
15:9	<b>Non-Posted Header Credit</b> Default advertised Non-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. x1 Port = 7 Header Credits x4 Port = 26 Header Credits x8 Port = 49 Header Credits	RWS	Yes	Refer to Description
17:16	<b>UpdateFC High-Priority Threshold for Non-Posted Payload Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details. 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> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details. 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
22:20	<b>Not used</b>	RWS	Yes	000b
23	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-406. A04h, A1Ch, A34h, A4Ch, A64h, A7Ch, A94h, AACh, AC4h, ADCh, AF4h, B0Ch, B24h, B3Ch, B54h, B6Ch INCH Threshold Port x VC0 Non-Posted (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
27:24	<b>BIST Port Control, Destination Port</b> Determines the Destination Half Station and Port, for TLPs received by a Port.	RWS	Yes	0h
28	<b>BIST Port Control, Start of Chain</b> Used to enable injection of a fixed Header and Payload in the corresponding Reader. 1 = Indicates that a Port is at the beginning of the test chain	RW	Yes	0
29	<b>BIST Port Control, End of Chain</b> Used to cause packets to be dropped in the corresponding ingress decoder. 1 = Indicates that a Port is at the end of the test chain	RW	Yes	0
30	<b>BIST Port Status, Port Header Error</b> 1 = Indicates that a Header (that is received by a Port) does not compare with the expected value. Writing a 1 Clears this bit.	RW1C	No	0
31	<b>BIST Port Status, Port Payload Error</b> 1 = Indicates that a Payload (that is received by a Port) does not compare with the expected value. Writing a 1 Clears this bit.	RW1C	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-407. A08h, A20h, A38h, A50h, A68h, A80h, A98h, AB0h, AC8h, AE0h, AF8h, B10h, B28h, B40h, B58h, B70h INCH Threshold Port x VC0 Completion (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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.				
Port x is Ports 0 through 15 for register offsets A08h, A20h, A38h, A50h, A68h, A80h, A98h, AB0h, AC8h, AE0h, AF8h, B10h, B28h, B40h, B58h, and B70h, respectively. (Refer to <a href="#">Table 14-44</a> .)				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<p><b>Completion Payload Credit</b>            Default advertised Completion Payload credit. Actual value is dependent upon Link width and Port configuration.            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.</p> <p>x1 Port = 64 Payload Credits            x4 Port = 224 Payload Credits            x8 Port = 320 Payload Credits</p>	RWS	Yes	Refer to Description
15:9	<p><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.</p> <p>x1 Port = 5 Header Credits            x4 Port = 26 Header Credits            x8 Port = 49 Header Credits</p>	RWS	Yes	Refer to Description
17:16	<p><b>UpdateFC High-Priority Threshold for Completion Payload Credit</b>            Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.</p> <p>00b = 75% (default)            01b = 50%            10b = 25%            11b = 100%</p>	RWS	Yes	00b
19:18	<p><b>UpdateFC High-Priority Threshold for Completion Header Credit</b>            Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.</p> <p>00b = 75% (default)            01b = 50%            10b = 25%            11b = 100%</p>	RWS	Yes	00b
22:20	<b>Not used</b>	RWS	Yes	000b
31:23	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-408. A0Ch, A24h, A3Ch, A54h, A6Ch, A84h, A9Ch, AB4h, ACCh, AE4h, AFCh, B14h, B2Ch, B44h, B5Ch, B74h INCH Threshold Port x VC1 Posted  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Posted credits are used for VC1 Memory Write and Message transactions. Port <i>x</i> is Ports 0 through 15 for register offsets A0Ch, A24h, A3Ch, A54h, A6Ch, A84h, A9Ch, AB4h, ACCh, AE4h, AFCh, B14h, B2Ch, B44h, B5Ch, and B74h, respectively. (Refer to <a href="#">Table 14-44</a> .)				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<p><b>Posted Payload Credit</b>            Default advertised Posted Payload credit. Actual value is dependent upon Link width and Port configuration.            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.            The PEX 8619 hardwires this field to 00h (infinite credits).</p> <p><b>Note:</b> <i>The default initial credit is infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed, using a serial EEPROM.</i></p>	RWS	Yes	00h
15:9	<p><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.            The PEX 8619 hardwires this field to 00h (infinite credits).</p> <p><b>Note:</b> <i>The default initial credit is infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed, using a serial EEPROM.</i></p>	RWS	Yes	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-408. A0Ch, A24h, A3Ch, A54h, A6Ch, A84h, A9Ch, AB4h, ACCh, AE4h, AFCh, B14h, B2Ch, B44h, B5Ch, B74h INCH Threshold Port x VC1 Posted (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details. 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for Posted Header Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details. 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. Refer to <a href="#">Section 9.3.5, “Credit Allocation When Common Pool Is Consumed,”</a> for further details. 000b = Request is weighted, based upon the Port’s Link width relative to the effective Link widths of the other Half Station’s 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-409. A10h, A28h, A40h, A58h, A70h, A88h, AA0h, AB8h, AD0h, AE8h, B00h, B18h, B30h, B48h, B60h, B78h INCH Threshold Port x VC1 Non-Posted (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Non-Posted credits are used for VC1 Memory Read, I/O Read, I/O Write, Configuration Read, and Configuration Write transactions. Port <i>x</i> is Ports 0 through 15 for register offsets A10h, A28h, A40h, A58h, A70h, A88h, AA0h, AB8h, AD0h, AE8h, B00h, B18h, B30h, B48h, B60h, and B78h, respectively. (Refer to <a href="#">Table 14-44</a> .)				
8:0	<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. Because of this, the PEX 8619 hardwires this field to 000h (infinite credits).	RsvdP	Yes	000h
15:9	<b>Non-Posted Header Credit</b> Default advertised Non-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. The PEX 8619 hardwires this field to 00h (infinite credits).  <i>Note:</i> <i>The default initial credit is infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed, using a serial EEPROM.</i>	RWS	Yes	00h
17:16	<b>UpdateFC High-Priority Threshold for Non-Posted Payload Credit</b> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  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> Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.  00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
22:20	<b>Not used</b>	RWS	Yes	000b
31:23	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-410. A14h, A2Ch, A44h, A5Ch, A74h, A8Ch, AA4h, ABCh, AD4h, AECh, B04h, B1Ch, B34h, B4Ch, B64h, B7Ch INCH Threshold Port x VC1 Completion (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Completion credits are used for VC1 Memory Read, I/O Read, I/O Write, Configuration Read, and Configuration Write transaction Completions.				
Port x is Ports 0 through 15 for register offsets A14h, A2Ch, A44h, A5Ch, A74h, A8Ch, AA4h, ABCh, AD4h, AECh, B04h, B1Ch, B34h, B4Ch, B64h, and B7Ch, respectively. (Refer to <a href="#">Table 14-44</a> .)				
2:0	<b>Reserved</b>	RsvdP	No	000b
8:3	<p><b>Completion Payload Credit</b>            Default advertised Completion Payload credit. Actual value is dependent upon Link width and Port configuration.            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.            The PEX 8619 hardwires this field to 00h (infinite credits).</p> <p><b>Note:</b> <i>The default initial credit is infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed, using a serial EEPROM.</i></p>	RWS	Yes	00h
15:9	<p><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. The PEX 8619 hardwires this field to 00h (infinite credits).</p> <p><b>Note:</b> <i>The default initial credit is infinite. If VC1 is enabled, the user must calculate and program the needed optimal VC1 initial credit values. These values should be programmed, using a serial EEPROM.</i></p>	RWS	Yes	00h
17:16	<p><b>UpdateFC High-Priority Threshold for Completion Payload Credit</b>            Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.</p> <p>00b = 75% (default)            01b = 50%            10b = 25%            11b = 100%</p>	RWS	Yes	00b
19:18	<p><b>UpdateFC High-Priority Threshold for Completion Header Credit</b>            Refer to <a href="#">Section 9.2.2, “UpdateFC DLLP Policy,”</a> for details.</p> <p>00b = 75% (default)            01b = 50%            10b = 25%            11b = 100%</p>	RWS	Yes	00b
22:20	<b>Not used</b>	RWS	Yes	000b
31:23	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.14 Device-Specific Registers – Physical Layer (Offsets B80h – C30h)

This section details the Device-Specific Physical Layer (PHY) registers located at offsets B80h through C30h. [Table 14-45](#) defines the register map.

Other Device-Specific PHY registers are detailed in:

- [Section 14.14.2, “Device-Specific Registers – Physical Layer \(Offsets 200h – 25Ch\)”](#)
- [Section 14.16.16, “Device-Specific Registers – Physical Layer \(Offsets E40h – EFCh\)”](#)

**Table 14-45. Device-Specific PHY Register Map  
(Offsets B80h – C30h) (Port 0, and also the NT Port  
Virtual Interface if Port 0 is a Legacy NT 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>	Even Ports – Loopback Master Status/Control	B80h
<i>Factory Test Only</i>	Odd Ports – Loopback Master Status/Control	B84h
	Even Ports 0, 2, 4, 6 – Receiver Error Counters	B88h
	Odd Ports 1, 3, 5, 7 – Receiver Error Counters	B8Ch
	Even Ports 8, 10, 12, 14 – Receiver Error Counters	B90h
	Odd Ports 9, 11, 13, 15 – Receiver Error Counters	B94h
	SerDes Drive Level 0	B98h
	SerDes Drive Level 1	B9Ch
	SerDes Drive Level 2	BA0h
	SerDes Drive Level 3	BA4h
	Post-Cursor Emphasis Level 0	BA8h
	Post-Cursor Emphasis Level 1	BACh
	Post-Cursor Emphasis Level 2	BB0h
	Post-Cursor Emphasis Level 3	BB4h
	Receiver Equalization Level 0	BB8h
	Receiver Equalization Level 1	BBCh
	Signal Detect Level	BC0h
<i>Factory Test Only</i>		BC4h – C24h
<i>SerDes Control</i>		C28h
<i>Factory Test Only</i>		C2Ch – C30h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-411. B80h Even Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Loopback Master Entry Failed</b> 1 = Indicates that Port 0 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 0 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[3]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
1	<b>Port 2 Loopback Master Entry Failed</b> 1 = Indicates that Port 2 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 2 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[7]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
2	<b>Port 4 Loopback Master Entry Failed</b> 1 = Indicates that Port 4 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 4 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[11]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
3	<b>Port 6 Loopback Master Entry Failed</b> 1 = Indicates that Port 6 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 6 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[15]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-411. B80h Even Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>Port 8 Loopback Master Entry Failed</b> 1 = Indicates that Port 8 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 8 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[19]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
5	<p><b>Port 10 Loopback Master Entry Failed</b> 1 = Indicates that Port 10 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 10 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[23]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
6	<p><b>Port 12 Loopback Master Entry Failed</b> 1 = Indicates that Port 12 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 12 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[27]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
7	<p><b>Port 14 Loopback Master Entry Failed</b> 1 = Indicates that Port 14 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 14 Even Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 220h[31]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-411. B80h Even Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 0 External Loopback Enable</b> 1 = Allows Port 0 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
9	<b>Port 2 External Loopback Enable</b> 1 = Allows Port 2 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
10	<b>Port 4 External Loopback Enable</b> 1 = Allows Port 4 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
11	<b>Port 6 External Loopback Enable</b> 1 = Allows Port 6 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
12	<b>Port 8 External Loopback Enable</b> 1 = Allows Port 8 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
13	<b>Port 10 External Loopback Enable</b> 1 = Allows Port 10 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
14	<b>Port 12 External Loopback Enable</b> 1 = Allows Port 12 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
15	<b>Port 14 External Loopback Enable</b> 1 = Allows Port 14 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-411. B80h Even Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>PRBS Pattern Sync Status Device Lane 0</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
17	<b>PRBS Pattern Sync Status Device Lane 1</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
18	<b>PRBS Pattern Sync Status Device Lane 2</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
19	<b>PRBS Pattern Sync Status Device Lane 3</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-411. B80h Even Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>PRBS Pattern Sync Status Device Lane 4</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
21	<b>PRBS Pattern Sync Status Device Lane 5</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
22	<b>PRBS Pattern Sync Status Device Lane 6</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
23	<b>PRBS Pattern Sync Status Device Lane 7</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
31:24	<b>Factory Test Only</b>	RWS	Yes	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-412. B84h Odd Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<p><b>Port 1 Loopback Master Entry Failed</b> 1 = Indicates that Port 1 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 1 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[3]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
1	<p><b>Port 3 Loopback Master Entry Failed</b> 1 = Indicates that Port 3 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 3 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[7]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
2	<p><b>Port 5 Loopback Master Entry Failed</b> 1 = Indicates that Port 5 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 5 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[11]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0
3	<p><b>Port 7 Loopback Master Entry Failed</b> 1 = Indicates that Port 7 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state</p> <p><i>Note:</i> If this bit and the Port 7 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[15]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.</p>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-412. B84h Odd Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 9 Loopback Master Entry Failed</b> 1 = Indicates that Port 9 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 9 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[19]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
5	<b>Port 11 Loopback Master Entry Failed</b> 1 = Indicates that Port 11 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 11 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[23]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
6	<b>Port 13 Loopback Master Entry Failed</b> 1 = Indicates that Port 13 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 13 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[27]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
7	<b>Port 15 Loopback Master Entry Failed</b> 1 = Indicates that Port 15 failed to enter the <i>Loopback</i> state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the <i>Detect</i> state <i>Note:</i> If this bit and the Port 15 Odd Port Physical Layer Command/Status register Port x Ready as Loopback Master bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 224h[31]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	RW1C	Yes	0
8	<b>Port 1 External Loopback Enable</b> 1 = Allows Port 1 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
9	<b>Port 3 External Loopback Enable</b> 1 = Allows Port 3 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
10	<b>Port 5 External Loopback Enable</b> 1 = Allows Port 5 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
11	<b>Port 7 External Loopback Enable</b> 1 = Allows Port 7 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-412. B84h Odd Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 9 External Loopback Enable</b> 1 = Allows Port 9 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
13	<b>Port 11 External Loopback Enable</b> 1 = Allows Port 11 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
14	<b>Port 13 External Loopback Enable</b> 1 = Allows Port 13 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
15	<b>Port 15 External Loopback Enable</b> 1 = Allows Port 15 to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when the Port's Receivers are directly connected, externally, to its Transmitters.	RW	Yes	0
16	<b>PRBS Pattern Sync Status Device Lane 8</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
17	<b>PRBS Pattern Sync Status Device Lane 9</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
18	<b>PRBS Pattern Sync Status Device Lane 10</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
19	<b>PRBS Pattern Sync Status Device Lane 11</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-412. B84h Odd Ports – Loopback Master Status/Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>PRBS Pattern Sync Status Device Lane 12</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
21	<b>PRBS Pattern Sync Status Device Lane 13</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
22	<b>PRBS Pattern Sync Status Device Lane 14</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
23	<b>PRBS Pattern Sync Status Device Lane 15</b> Diagnoses broken Loopback wiring, because no errors can be detected and counted until pattern sync is detected. Sync achieved is Active-High. 0 = Causes the PRBS pattern generator, when enabled, to output the PRBS7 sequence 1 = Indicates that the corresponding Lane's PRBS Data Checker has synchronized to the incoming PRBS data pattern – the Receiver has acquired its first match on two sequentially received words	RO	No	0
31:24	<b>Factory Test Only</b>	RWS	Yes	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-413. B88h Even Ports 0, 2, 4, 6 – Receiver Error Counters  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Port 0 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
15:8	<b>Port 2 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
23:16	<b>Port 4 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
31:24	<b>Port 6 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h

**Register 14-414. B8Ch Odd Ports 1, 3, 5, 7 – Receiver Error Counters  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Port 1 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
15:8	<b>Port 3 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
23:16	<b>Port 5 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
31:24	<b>Port 7 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-415. B90h Even Ports 8, 10, 12, 14 – Receiver Error Counters  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Port 8 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
15:8	<b>Port 10 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
23:16	<b>Port 12 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
31:24	<b>Port 14 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h

**Register 14-416. B94h Odd Ports 9, 11, 13, 15 – Receiver Error Counters  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Port 9 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
15:8	<b>Port 11 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
23:16	<b>Port 13 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h
31:24	<b>Port 15 Receiver Error Counter</b> When read, returns the quantity of Receiver errors detected by the corresponding Port. 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.	RW1C	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-417. B98h SerDes Drive Level 0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
	The default value of this register, when combined with the <b>Post-Cursor Emphasis Level 0</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>B8h</b> ) default value, provides -3.5 dB of de-emphasis. However, it is also a Status register with provisional read-back data.  The power-up reset default values of this register correspond to -3.5 dB drive levels, and can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <b>current value being applied to the Lane</b> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.			
	<b>Note:</b> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the Post-Cursor Emphasis Level 0 register.			
4:0	<b>SerDes 0 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 0 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 1 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 1 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 2 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 2 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 3 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 3 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-418. B9Ch SerDes Drive Level 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
	The default value of this register, when combined with the <b>Post-Cursor Emphasis Level 1</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>BACh</b> ) default value, provides -3.5 dB of de-emphasis. However, it is also a Status register with provisional read-back data.  The power-up reset default values of this register correspond to -3.5 dB drive levels, and can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <b>current value being applied to the Lane</b> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.			
	<b>Note:</b> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>Post-Cursor Emphasis Level 1</b> register.			
4:0	<b>SerDes 4 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 4 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 5 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 5 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 6 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 6 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 7 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset <b>98h[6]</b> , is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 7 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-419. BA0h SerDes Drive Level 2  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The default value of this register, when combined with the <b>Post-Cursor Emphasis Level 2</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset BB0h) default value, provides -3.5 dB of de-emphasis. However, it is also a Status register with provisional read-back data.				
The power-up reset default values of this register correspond to -3.5 dB drive levels, and can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <i>current value being applied to the Lane</i> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.				
<i>Note:</i> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the Post-Cursor Emphasis Level 2 register.				
4:0	<b>SerDes 8 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 8 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 9 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 9 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 10 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 10 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 11 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 db)  0Eh (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 11 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-420. BA4h SerDes Drive Level 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
	The default value of this register, when combined with the <b>Post-Cursor Emphasis Level 3</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset BB4h) default value, provides -3.5 dB of de-emphasis. However, it is also a Status register with provisional read-back data.  The power-up reset default values of this register correspond to -3.5 dB drive levels, and can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <b>current value being applied to the Lane</b> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.			
	<b>Note:</b> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>Post-Cursor Emphasis Level 3</b> register.			
4:0	<b>SerDes 12 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 12 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 13 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 13 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 14 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 14 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 15 Drive Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <b>Selectable De-Emphasis</b> bit, offset 98h[6], is Cleared, provided that the <b>Current Link Speed</b> is 5.0 GT/s (Port’s <b>Link Status</b> register <b>Current Link Speed</b> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 0Eh.	RWS	Yes	10h (2.5 GT/s, -3.5 dB)  0Eh (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 15 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-421. BA8h Post-Cursor Emphasis Level 0  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The default value of this register, when combined with the <b>SerDes Drive Level 0</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset B98h) default value, provides -3.5 dB of de-emphasis.				
The power-up value can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <b>current value being applied to the Lane</b> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.				
<i>Note:</i> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>SerDes Drive Level 0</b> register.				
4:0	<b>SerDes 0 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 0 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 1 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 1 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 2 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 2 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 3 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 3 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-422. BAC<sub>h</sub> Post-Cursor Emphasis Level 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The default value of this register, when combined with the <b>SerDes Drive Level 1</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset B9Ch) default value, provides -3.5 dB of de-emphasis.				
The power-up value can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <i>current value being applied to the Lane</i> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.				
<i>Note:</i> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>SerDes Drive Level 1</b> register.				
4:0	<b>SerDes 4 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 4 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 5 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 5 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 6 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 6 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 7 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 7 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-423. BB0h Post-Cursor Emphasis Level 2  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The default value of this register, when combined with the <b>SerDes Drive Level 2</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>BA0h</b> ) default value, provides -3.5 dB of de-emphasis.				
The power-up value can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <b>current value being applied to the Lane</b> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.				
<i>Note:</i> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>SerDes Drive Level 2</b> register.				
4:0	<b>SerDes 8 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 8 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 9 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 9 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 10 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 10 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 11 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port's <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset <b>98h[6]</b> , is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port's <b>Link Status</b> register <i>Current Link Speed</i> field, offset <b>78h[19:16]</b> , is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 db)  15h (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 11 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-424. BB4h Post-Cursor Emphasis Level 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The default value of this register, when combined with the <b>SerDes Drive Level 3</b> register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>BA4h</b> ) default value, provides -3.5 dB of de-emphasis.				
The power-up value can be overwritten by serial EEPROM and/or Configuration Space register transactions. However, when read back, the returned data always represents the <i>current value being applied to the Lane</i> . That is because the Write Data value is written into a holding register for the -3.5 dB value, but cannot be directly read from the register. The Read Data value is always sourced from an active Drive register that is loaded dynamically upon de-emphasis changes.				
<i>Note:</i> Refer to <a href="#">Section 21.7, “Transmit Drive Characteristics,”</a> for a complete breakout of the default values, and their relationship with the <b>SerDes Drive Level 3</b> register.				
4:0	<b>SerDes 12 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<b>SerDes 12 Auto Load Disable</b>	RWS	Yes	0
12:8	<b>SerDes 13 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
14:13	<b>Reserved</b>	RsvdP	No	00b
15	<b>SerDes 13 Auto Load Disable</b>	RWS	Yes	0
20:16	<b>SerDes 14 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
22:21	<b>Reserved</b>	RsvdP	No	00b
23	<b>SerDes 14 Auto Load Disable</b>	RWS	Yes	0
28:24	<b>SerDes 15 Post-Cursor Emphasis Level</b> If 5.0 GT/s -6 dB is selected (Port’s <b>Link Control 2</b> register <i>Selectable De-Emphasis</i> bit, offset 98h[6], is Cleared, provided that the Current Link Speed is 5.0 GT/s (Port’s <b>Link Status</b> register <i>Current Link Speed</i> field, offset 78h[19:16], is programmed to 0010b)), this register is automatically loaded with default value 15h.	RWS	Yes	0Dh (2.5 GT/s, -3.5 dB)  15h (5.0 GT/s, -6 dB)
30:29	<b>Reserved</b>	RsvdP	No	00b
31	<b>SerDes 15 Auto Load Disable</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-425. BB8h Receiver Equalization Level 0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
--------	-------------	------	------------------------------------	---------

This register provides the Receiver Equalization Level control for the lower SerDes.

Rx Equalization[3:0]	Equalization	Rx Equalization[3:0]	Equalization
0h (default)	Off	7h to 9h	Medium
1h	Minimum	Ah to Dh	High to Medium
2h to 3h	Low	Eh to Fh	Maximum
4h to 6h	Low to Medium		

3:0	<b>SerDes 0 Receiver Equalization Level</b>	RWS	Yes	0h
7:4	<b>SerDes 1 Receiver Equalization Level</b>	RWS	Yes	0h
11:8	<b>SerDes 2 Receiver Equalization Level</b>	RWS	Yes	0h
15:12	<b>SerDes 3 Receiver Equalization Level</b>	RWS	Yes	0h
19:16	<b>SerDes 4 Receiver Equalization Level</b>	RWS	Yes	0h
23:20	<b>SerDes 5 Receiver Equalization Level</b>	RWS	Yes	0h
27:24	<b>SerDes 6 Receiver Equalization Level</b>	RWS	Yes	0h
31:28	<b>SerDes 7 Receiver Equalization Level</b>	RWS	Yes	0h

**Register 14-426. BBCh Receiver Equalization Level 1**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
--------	-------------	------	------------------------------------	---------

This register provides the Receiver Equalization Level control for the upper SerDes.

Rx Equalization[3:0]	Equalization	Rx Equalization[3:0]	Equalization
0h (default)	Off	7h to 9h	Medium
1h	Minimum	Ah to Dh	High to Medium
2h to 3h	Low	Eh to Fh	Maximum
4h to 6h	Low to Medium		

3:0	<b>SerDes 8 Receiver Equalization Level</b>	RWS	Yes	0h
7:4	<b>SerDes 9 Receiver Equalization Level</b>	RWS	Yes	0h
11:8	<b>SerDes 10 Receiver Equalization Level</b>	RWS	Yes	0h
15:12	<b>SerDes 11 Receiver Equalization Level</b>	RWS	Yes	0h
19:16	<b>SerDes 12 Receiver Equalization Level</b>	RWS	Yes	0h
23:20	<b>SerDes 13 Receiver Equalization Level</b>	RWS	Yes	0h
27:24	<b>SerDes 14 Receiver Equalization Level</b>	RWS	Yes	0h
31:28	<b>SerDes 15 Receiver Equalization Level</b>	RWS	Yes	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-427. BC0h Signal Detect Level  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides the Receiver Signal Detect Level select. Each two-bit field in this register provides four Settings for detecting Electrical Idle Analog, for the corresponding SerDes.				
00b	Approximately 50 to 80 mV			
01b	Approximately 65 to 175 mV (default)			
10b	Approximately 75 to 200 mV			
11b	Approximately 120 to 240 mV			
1:0	<b>SerDes 0 Signal Detect Level</b>	RWS	Yes	01b
3:2	<b>SerDes 1 Signal Detect Level</b>	RWS	Yes	01b
5:4	<b>SerDes 2 Signal Detect Level</b>	RWS	Yes	01b
7:6	<b>SerDes 3 Signal Detect Level</b>	RWS	Yes	01b
9:8	<b>SerDes 4 Signal Detect Level</b>	RWS	Yes	01b
11:10	<b>SerDes 5 Signal Detect Level</b>	RWS	Yes	01b
13:12	<b>SerDes 6 Signal Detect Level</b>	RWS	Yes	01b
15:14	<b>SerDes 7 Signal Detect Level</b>	RWS	Yes	01b
17:16	<b>SerDes 8 Signal Detect Level</b>	RWS	Yes	01b
19:18	<b>SerDes 9 Signal Detect Level</b>	RWS	Yes	01b
21:20	<b>SerDes 10 Signal Detect Level</b>	RWS	Yes	01b
23:22	<b>SerDes 11 Signal Detect Level</b>	RWS	Yes	01b
25:24	<b>SerDes 12 Signal Detect Level</b>	RWS	Yes	01b
27:26	<b>SerDes 13 Signal Detect Level</b>	RWS	Yes	01b
29:28	<b>SerDes 14 Signal Detect Level</b>	RWS	Yes	01b
31:30	<b>SerDes 15 Signal Detect Level</b>	RWS	Yes	01b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-428. C28h SerDes Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register controls SerDes logic parameters.				
2:0	<b>Receiver Detect Time Select</b> Selects the Receiver Detect timing. 000b = 1.0 µs 001b = 2.0 µs 010b = 4.0 µs (default) 011b = 5.0 µs 100b = 10.0 µs 101b = 20.0 µs 110b = 40.0 µs 111b = 50.0 µs	RWS	Yes	010b
3	<b>Reserved</b>	RsvdP	No	0
7:4	<b>Factory Test Only</b>	RWS	Yes	Fh
15:8	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-428. C28h SerDes Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<p><b>Port 0 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
17	<p><b>Port 1 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
18	<p><b>Port 2 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
19	<p><b>Port 3 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-428. C28h SerDes Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<p><b>Port 4 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
21	<p><b>Port 5 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
22	<p><b>Port 6 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
23	<p><b>Port 7 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-428. C28h SerDes Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>Port 8 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
25	<p><b>Port 9 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
26	<p><b>Port 10 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
27	<p><b>Port 11 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-428. C28h SerDes Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>Port 12 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
29	<p><b>Port 13 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
30	<p><b>Port 14 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Even Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 200h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0
31	<p><b>Port 15 Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 204h[15:8]) are Set, for the SerDes associated with the Port. 1 = Overall Electrical Idle inference logic is disabled on the corresponding Port. Electrical Idle inference during the <i>Recovery.Speed</i> state is not affected and will continue to operate.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 14.16.15 Device-Specific Registers – Port Configuration Header (Offsets E00h – E3Ch)

This section details the Device-Specific Port Configuration Header registers, for each Port. [Table 14-46](#) defines the register map.

**Table 14-46. Device-Specific Port Configuration Header Register Map (Port 0)**

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 Command Port 1	PCI Command Port 0	E00h
PCI Command Port 3	PCI Command Port 2	E04h
PCI Command Port 5	PCI Command Port 4	E08h
PCI Command Port 7	PCI Command Port 6	E0Ch
PCI Command Port 9	PCI Command Port 8	E10h
PCI Command Port 11	PCI Command Port 10	E14h
PCI Command Port 13	PCI Command Port 12	E18h
PCI Command Port 15	PCI Command Port 14	E1Ch
Bridge Control Port 1	Bridge Control Port 0	E20h
Bridge Control Port 3	Bridge Control Port 2	E24h
Bridge Control Port 5	Bridge Control Port 4	E28h
Bridge Control Port 7	Bridge Control Port 6	E2Ch
Bridge Control Port 9	Bridge Control Port 8	E30h
Bridge Control Port 11	Bridge Control Port 10	E34h
Bridge Control Port 13	Bridge Control Port 12	E38h
Bridge Control Port 15	Bridge Control Port 14	E3Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-429. E00h PCI Command Port 0  
(Port 0)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 0's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 0's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 0's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 0's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream.  0 = PEX 8619 handles Memory and I/O Requests received on Port 0's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-430. E02h PCI Command Port 1  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 1's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 1's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 1's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 1's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 1's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-431. E04h PCI Command Port 2  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 2's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 2's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 2's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 2's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 2's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-432. E06h PCI Command Port 3  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 3's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 3's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 3's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 3's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 3's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-433. E08h PCI Command Port 4  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 4's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 4's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 4's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 4's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 4's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-434. E0Ah PCI Command Port 5  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 5's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 5's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 5's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 5's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 5's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-435. E0Ch PCI Command Port 6  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 6's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 6's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 6's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 6's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 6's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-436. E0Eh PCI Command Port 7  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 7's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 7's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 7's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 7's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 7's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-437. E10h PCI Command Port 8  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 8's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 8's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 8's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 8's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 8's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-438. E12h PCI Command Port 9  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 9's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 9's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 9's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 9's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 9's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-439. E14h PCI Command Port 10  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 10's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 10's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 10's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 10's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 10's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-440. E16h PCI Command Port 11  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 11's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 11's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 11's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 11's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 11's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-441. E18h PCI Command Port 12  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 12's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 12's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 12's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 12's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 12's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-442. E1Ah PCI Command Port 13  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 13's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 13's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 13's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 13's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 13's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-443. E1Ch PCI Command Port 14  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 14's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 14's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 14's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 14's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 14's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-444. E1Eh PCI Command Port 15  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>I/O Access Enable</b> 0 = PEX 8619 ignores I/O Space accesses on Port 15's primary interface 1 = PEX 8619 responds to I/O Space accesses on Port 15's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on Port 15's primary interface 1 = PEX 8619 responds to Memory Space accesses on Port 15's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 Memory and I/O Request forwarding upstream. Neither affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling upstream or downstream. 0 = PEX 8619 handles Memory and I/O Requests received on Port 15's downstream/secondary interface as URs; for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory and I/O Requests upstream	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>SERR# Enable</b> Controls the <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]). 0 = No error is detected 1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex	RW	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### Register 14-445. E20h Bridge Control Port 0 (Port 0)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<p><b>SERR# Enable</b></p> <p>Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface.</p> <p>When Set, and the <b>PCI Command Port 0</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.</p>	RW	Yes	0
2	<p><b>ISA Enable</b></p> <p>Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh).</p> <p>When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0.</p> <p>0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers</p> <p>1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)</p>	RW	Yes	0
3	<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 2 (<i>ISA Enable</i>) Setting</li> </ul> <p>Forwarding of these addresses is qualified by the <b>PCI Command Port 0</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. 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>I/O Access Enable</i> and <i>Memory 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>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-445. E20h Bridge Control Port 0  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-446. E22h Bridge Control Port 1  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 1</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<b>ISA Enable</b>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 1</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-446. E22h Bridge Control Port 1  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-447. E24h Bridge Control Port 2  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<p><b>SERR# Enable</b></p> <p>Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface.</p> <p>When Set, and the <b>PCI Command Port 2</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.</p>	RW	Yes	0
2	<p><b>ISA Enable</b></p> <p>Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh).</p> <p>When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0.</p> <p>0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers</p> <p>1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)</p>	RW	Yes	0
3	<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 2 (<i>ISA Enable</i>) Setting</li> </ul> <p>Forwarding of these addresses is qualified by the <b>PCI Command Port 2</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. 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>I/O Access Enable</i> and <i>Memory 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>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-447. E24h Bridge Control Port 2  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-448. E26h Bridge Control Port 3  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 3</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 3</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-448. E26h Bridge Control Port 3  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-449. E28h Bridge Control Port 4  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 4</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<b>ISA Enable</b>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 4</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-449. E28h Bridge Control Port 4  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-450. E2Ah Bridge Control Port 5  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 5</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<b>ISA Enable</b>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 5</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-450. E2Ah Bridge Control Port 5  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### Register 14-451. E2Ch Bridge Control Port 6 (Port 0)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<p><b>SERR# Enable</b></p> <p>Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface.</p> <p>When Set, and the <b>PCI Command Port 6</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.</p>	RW	Yes	0
2	<p><b>ISA Enable</b></p> <p>Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh).</p> <p>When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0.</p> <p>0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers</p> <p>1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)</p>	RW	Yes	0
3	<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 2 (<b>ISA Enable</b>) Setting</li> </ul> <p>Forwarding of these addresses is qualified by the <b>PCI Command Port 6</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. 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 <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-451. E2Ch Bridge Control Port 6  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-452. E2Eh Bridge Control Port 7  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<p><b>SERR# Enable</b></p> <p>Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface.</p> <p>When Set, and the <b>PCI Command Port 7</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.</p>	RW	Yes	0
2	<p><b>ISA Enable</b></p> <p>Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh).</p> <p>When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0.</p> <p>0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers</p> <p>1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)</p>	RW	Yes	0
3	<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 2 (<b>ISA Enable</b>) Setting</li> </ul> <p>Forwarding of these addresses is qualified by the <b>PCI Command Port 7</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. 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 <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-452. E2Eh Bridge Control Port 7  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-453. E30h Bridge Control Port 8  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 8</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<b>ISA Enable</b>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 8</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-453. E30h Bridge Control Port 8  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VGA 16-Bit Enable</b> Used only when bit 3 ( <a href="#">VGA Enable</a> ) 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	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-454. E32h Bridge Control Port 9  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<p><b>SERR# Enable</b></p> <p>Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface.</p> <p>When Set, and the <b>PCI Command Port 9</b> register <b>SERR# Enable</b> bit is Set, enables the <b>PCI Status</b> register <b>Signaled System Error</b> bit.</p>	RW	Yes	0
2	<p><b>ISA Enable</b></p> <p>Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh).</p> <p>When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0.</p> <p>0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers</p> <p>1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)</p>	RW	Yes	0
3	<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 2 (<b>ISA Enable</b>) Setting</li> </ul> <p>Forwarding of these addresses is qualified by the <b>PCI Command Port 9</b> register <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits. 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 <b>I/O Access Enable</b> and <b>Memory Access Enable</b> bits are Set), independent of the Memory and I/O Address ranges and independent of the <b>ISA Enable</b> bit</p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-454. E32h Bridge Control Port 9  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-455. E34h Bridge Control Port 10  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 10</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 10</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-455. E34h Bridge Control Port 10  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-456. E36h Bridge Control Port 11  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 11</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 11</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-456. E36h Bridge Control Port 11  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-457. E38h Bridge Control Port 12  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 12</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 12</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-457. E38h Bridge Control Port 12  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-458. E3Ah Bridge Control Port 13  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 13</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 13</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-458. E3Ah Bridge Control Port 13  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-459. E3Ch Bridge Control Port 14  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 14</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 14</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-459. E3Ch Bridge Control Port 14  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<i>VGA Enable</i>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-460. E3Eh Bridge Control Port 15  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b>	RsvdP	No	0
1	<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 <b>PCI Command Port 15</b> register <i>SERR# Enable</i> bit is Set, enables the <b>PCI Status</b> register <i>Signaled System Error</i> bit.	RW	Yes	0
2	<b>ISA Enable</b> Modifies the bridge response to ISA I/O addresses. Applies only to I/O addresses, enabled by the <b>I/O Base</b> and <b>I/O Limit</b> registers, that are within the first 64 KB of PCI I/O Address space (0000_0000h to 0000_FFFFh). When Set, the bridge blocks forwarding of I/O transactions from the primary to secondary interface that address the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary interface), I/O transactions are forwarded if they address the last 768 bytes in each 1-KB block. The default state of this bit after reset must be 0. 0 = Forward downstream all I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers 1 = Forward upstream ISA I/O addresses in the Address range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers that are within the first 64 KB of PCI I/O Address space (top 768 bytes of each 1-KB block)	RW	Yes	0
3	<b>VGA Enable</b> Modifies the bridge response to VGA-compatible addresses. 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): <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> Additionally, when Set, forwarding of these addresses is independent of the: <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 2 (<i>ISA Enable</i>) Setting</li> </ul> Forwarding of these addresses is qualified by the <b>PCI Command Port 15</b> register <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits. The default state of this bit after reset must be 0. 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 1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>I/O Access Enable</i> and <i>Memory Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-460. E3Eh Bridge Control Port 15  
(Port 0) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>VGA 16-Bit Enable</b> Used only when bit 3 (<a href="#">VGA Enable</a>) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0.</p> <p>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.</p> <p>0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses</p>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 14.16.16 Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)

This section details the Device-Specific Physical Layer (PHY) registers located at offsets E40h through EFCh. [Table 14-47](#) defines the register map.

Other Device-Specific PHY registers are detailed in:

- [Section 14.14.2, “Device-Specific Registers – Physical Layer \(Offsets 200h – 25Ch\)”](#)
- [Section 14.16.14, “Device-Specific Registers – Physical Layer \(Offsets B80h – C30h\)”](#)

**Table 14-47. Device-Specific PHY Register Map  
(Offsets E40h – EFCh)**

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	
	CSR-Based SerDes Mode	E40h
	Internal Output A Trigger State Change Storage Mask	E44h
	Internal Output B Trigger State Change Storage Mask	E48h
	<i>Reserved</i>	E4Ch – EECh
	Physical Layer Control	EF0h
	<i>Reserved</i>	EF4h – EFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-461. E40h CSR-Based SerDes Mode  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>SerDes Mode Enable</b>	RW	Yes	0
3:1	<b>SerDes Station</b>	RW	Yes	000b
7:4	<b>SerDes Lane</b>	RW	Yes	0h
10:8	<b>SerDes Lane2</b>	RW	Yes	0h
11	<b>SerDes Lane Polarity</b>	RW	Yes	0
31:12	<i>Reserved</i>	RsvdP	No	0000_0h

**Register 14-462. E44h Internal Output A Trigger State Change Storage Mask  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:0	<b>Internal Output A Trigger State Change Mask</b>	RW	Yes	0-0h
31:18	<i>Reserved</i>	RsvdP	No	0-0h

**Register 14-463. E48h Internal Output B Trigger State Change Storage Mask  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:0	<b>Internal Output B Trigger State Change Mask</b>	RW	Yes	0-0h
31:18	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-464. EF0h Physical Layer Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register contains the parameters for controlling Electrical Idle Inference, and other optional training sequence controls.				
0	<b>SSC RXFIFO Skew Check Disable</b> 1 = Disables Rx FIFO SKIP addition, when rxvalids are skewed between FIFOs within a SerDes quad. This control is applicable only during a Spread-Spectrum Clock (SSC)-enabled chip operation.	RWS	Yes	0
1	<b>SSC RXFIFO Skew Counter Threshold</b> Selects how many clocks that a SKIP addition is blocked while rxvalids are skewed. 0 = 32 cycles 1 = 64 cycles	RWS	Yes	0
2	<b>SSC RXFIFO PhyStatus Mode Select</b> Internal PHY Interface for the PCI Express architecture's (PIPE) Physical Layer Status (PhyStatus) signals are Lane-independent. 1 = PIPE PhyStatus on all Lanes, of each SerDes quad, will be copies of Lane 0 from that quad <i>Note:</i> With SSC x8 operation, only SerDes Quads 0 and 1 are used. For SSC x4, only SerDes Quad 0 is used.	RWS	Yes	0
3	<b>Symbol Framer Reset Release Time Select</b> Selects the quantity of clocks for release from LX_RX_SIGDET becoming true. 0 = 8-clock delay 1 = 64-clock delay	RWS	Yes	0
7:4	<b>Factory Test Only</b>	RWS	Yes	0-0h
9:8	<b>Inferred Electrical Idle Exit Time Select Code</b> When Electrical Idle has been inferred and the Electrical Idle Inference Exit Type is 1, this field selects the amount of time that the SerDes Receive Data path remains disabled. 00b = 2 µs 01b = 4 µs 10b = 8 µs 11b = 16 µs	RWS	Yes	00b
11:10	<b>Reserved</b>	RsvdP	No	00b
13:12	<b>Symbol Framer Detection Time Select Code</b> When Electrical Idle has been inferred and the Electrical Idle Inference Exit Type is 1, this field selects the amount of time that the symbol framer is allowed to obtain symbol lock. 00b = 128 ns 01b = 256 ns 10b = 512 ns 11b = 1 µs	RWS	Yes	00b
15:14	<b>Reserved</b>	RsvdP	No	00b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-464. EF0h Physical Layer Control  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<p><b>Electrical Idle Inference on EIOS Receipt Enable</b> Electrical Idle Inference on Electrical Idle Ordered-Set (EIOS) Receipt enable, for the corresponding Port.</p> <p>0 = Electrical Idle inference is enabled upon EIOS receipt, if the <b>Even/Odd Port Electrical Idle for Compliance Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 200h[15:8], and Odd Ports, offset 204h[15:8]) are Set, for the SerDes associated with the Port</p> <p>1 = Electrical Idle will be inferred as soon as an EIOS is received on any Lane of the corresponding Port</p>	RWS	Yes	0
17	<p><b>Inferred Electrical Idle Inference Exit Type</b> Selects the method used to detect exit from Electrical Idle, after Electrical Idle has been inferred.</p> <p>0 = <b>Fast Method</b> – Type 0 Exit mode is used, which uses conventional analog Electrical Idle Exit Detection circuitry</p> <p>1 = <b>Slow Method</b> – Type 1 Exit mode is used, which uses the Symbol Framer Detection Time Select Code and Inferred Electrical Idle Exit Time Select Code Timers (fields [13:12] and 9:8], respectively)</p>	RWS	Yes	0
31:18	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 14.16.17 Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch)

This section details the Device-Specific Source Queue Weight and Soft Error registers. [Table 14-48](#) defines the register map.

**Table 14-48. Device-Specific Source Queue Weight and Soft Error Register Map (Offsets F00h – F3Ch)**

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>ECRC Check Enable for Ports 0, 1, 2, 3</b>
		<b>ECRC Check Enable for Ports 4, 5, 6, 7</b>
		<b>ECRC Check Enable for Ports 8, 9, 10, 11</b>
		<b>ECRC Check Enable for Ports 12, 13, 14, 15</b>
		<b>Port Egress TLP Threshold</b>
		<b>Reserved</b>
		<b>Soft Error Counters 1</b>
		<b>Reserved</b>
		F1Ch – F20h
<b>Reserved</b>		<b>Soft Error Counters 4</b>
<b>Reserved</b>		<b>Soft Error Counters 5</b>
		<b>Soft Error Counters 6</b>
		<b>Soft Error Injection</b>
		<b>Reserved</b>
		F34h – F3Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-465. F00h ECRC Check Enable for Ports 0, 1, 2, 3  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Port 0 registers at offsets F00h through F08h are a shadow copy of each Port's <b>Advanced Error Capabilities and Control</b> register <b>ECRC Check Enable</b> bit (offset FCCh[8]). To enable or disable ECRC Check functionality, use the <b>Advanced Error Capabilities and Control</b> register for each Port. Software should not program the shadow registers.				
0	<b>ECRC Check Enable for Port 0</b> 0 = ECRC checking is disabled on Port 0 1 = ECRC checking is enabled on Port 0	RWS	Yes	0
7:1	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>ECRC Check Enable for Port 1</b> 0 = ECRC checking is disabled on Port 1 1 = ECRC checking is enabled on Port 1	RWS	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>ECRC Check Enable for Port 2</b> 0 = ECRC checking is disabled on Port 2 1 = ECRC checking is enabled on Port 2	RWS	Yes	0
23:17	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>ECRC Check Enable for Port 3</b> 0 = ECRC checking is disabled on Port 3 1 = ECRC checking is enabled on Port 3	RWS	Yes	0
31:25	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-466. F04h ECRC Check Enable for Ports 4, 5, 6, 7  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Port 0 registers at offsets F00h through F08h are a shadow copy of each Port's <b>Advanced Error Capabilities and Control</b> register <b>ECRC Check Enable</b> bit (offset FCCh[8]). To enable or disable ECRC Check functionality, use the <b>Advanced Error Capabilities and Control</b> register for each Port. Software should not program the shadow registers.				
0	<b>ECRC Check Enable for Port 4</b> 0 = ECRC checking is disabled on Port 4 1 = ECRC checking is enabled on Port 4	RWS	Yes	0
7:1	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>ECRC Check Enable for Port 5</b> 0 = ECRC checking is disabled on Port 5 1 = ECRC checking is enabled on Port 5	RWS	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>ECRC Check Enable for Port 6</b> 0 = ECRC checking is disabled on Port 6 1 = ECRC checking is enabled on Port 6	RWS	Yes	0
23:17	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>ECRC Check Enable for Port 7</b> 0 = ECRC checking is disabled on Port 7 1 = ECRC checking is enabled on Port 7	RWS	Yes	0
31:25	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-467. F08h ECRC Check Enable for Ports 8, 9, 10, 11  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Port 0 registers at offsets F00h through F08h are a shadow copy of each Port's <b>Advanced Error Capabilities and Control</b> register <b>ECRC Check Enable</b> bit (offset FCCh[8]). To enable or disable ECRC Check functionality, use the <b>Advanced Error Capabilities and Control</b> register for each Port. Software should not program the shadow registers.				
0	<b>ECRC Check Enable for Port 8</b> 0 = ECRC checking is disabled on Port 8 1 = ECRC checking is enabled on Port 8	RWS	Yes	0
7:1	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>ECRC Check Enable for Port 9</b> 0 = ECRC checking is disabled on Port 9 1 = ECRC checking is enabled on Port 9	RWS	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>ECRC Check Enable for Port 10</b> 0 = ECRC checking is disabled on Port 10 1 = ECRC checking is enabled on Port 10	RWS	Yes	0
23:17	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>ECRC Check Enable for Port 11</b> 0 = ECRC checking is disabled on Port 11 1 = ECRC checking is enabled on Port 11	RWS	Yes	0
31:25	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-468. F0Ch ECRC Check Enable for Ports 12, 13, 14, 15  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The Port 0 registers at offsets F00h through F08h are a shadow copy of each Port's Advanced Error Capabilities and Control register ECRC Check Enable bit (offset FCCh[8]). To enable or disable ECRC Check functionality, use the Advanced Error Capabilities and Control register for each Port. Software should not program the shadow registers.				
0	<b>ECRC Check Enable for Port 12</b> 0 = ECRC checking is disabled on Port 12 1 = ECRC checking is enabled on Port 12	RWS	Yes	0
7:1	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>ECRC Check Enable for Port 13</b> 0 = ECRC checking is disabled on Port 13 1 = ECRC checking is enabled on Port 13	RWS	Yes	0
15:9	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>ECRC Check Enable for Port 14</b> 0 = ECRC checking is disabled on Port 14 1 = ECRC checking is enabled on Port 14	RWS	Yes	0
23:17	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>ECRC Check Enable for Port 15</b> 0 = ECRC checking is disabled on Port 15 1 = ECRC checking is enabled on Port 15	RWS	Yes	0
31:25	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-469. F10h Port Egress TLP Threshold  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Caution: Source Queuing and Read Pacing should not be concurrently enabled. The two features are incompatible and doing so can result in Fatal errors.</i>				
10:0	<b>Port Lower TLP Counter</b> When Source Scheduling is disabled due to the Port Upper TLP Counter (threshold) being exceeded, Source Scheduling is re-enabled when the Port TLP Counter goes below the Port Lower TLP Counter (this threshold). Because the default Setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8619, the Source Scheduler is disabled, by default.	RWS	Yes	7FFh
15:11	<b>Reserved</b>	RsvdP	No	0-0h
26:16	<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. Because the default Setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8619, the Source Scheduler is disabled, by default.	RWS	Yes	7FFh
31:27	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-470. F18h Soft Error Counters 1  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
1	<b>Port 2 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
2	<b>Port 4 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
3	<b>Port 6 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
4	<b>Port 8 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
5	<b>Port 10 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
6	<b>Port 12 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
7	<b>Port 14 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
8	<b>Port 0 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
9	<b>Port 2 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
10	<b>Port 4 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
11	<b>Port 6 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
12	<b>Port 8 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
13	<b>Port 10 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
14	<b>Port 12 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
15	<b>Port 14 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
16	<b>Port 1 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
17	<b>Port 3 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
18	<b>Port 5 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
19	<b>Port 7 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
20	<b>Port 9 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
21	<b>Port 11 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
22	<b>Port 13 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
23	<b>Port 15 Payload RAM 0 1-Bit Soft Error Counter Value</b>	RO	No	0
24	<b>Port 1 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
25	<b>Port 3 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
26	<b>Port 5 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
27	<b>Port 7 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
28	<b>Port 9 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
29	<b>Port 11 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
30	<b>Port 13 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0
31	<b>Port 15 Payload RAM 1 1-Bit Soft Error Counter Value</b>	RO	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-471. F24h Soft Error Counters 4  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
1	<b>Port 2 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
2	<b>Port 4 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
3	<b>Port 6 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
4	<b>Port 8 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
5	<b>Port 10 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
6	<b>Port 12 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
7	<b>Port 14 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
8	<b>Port 1 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
9	<b>Port 3 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
10	<b>Port 5 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
11	<b>Port 7 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
12	<b>Port 9 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
13	<b>Port 11 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
14	<b>Port 13 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
15	<b>Port 15 Header RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-472. F28h Soft Error Counters 5  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
1	<b>Port 2 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
2	<b>Port 4 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
3	<b>Port 6 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
4	<b>Port 8 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
5	<b>Port 10 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
6	<b>Port 12 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
7	<b>Port 14 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
8	<b>Port 1 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
9	<b>Port 3 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
10	<b>Port 5 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
11	<b>Port 7 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
12	<b>Port 9 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
13	<b>Port 11 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
14	<b>Port 13 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
15	<b>Port 15 Payload Linked List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-473. F2Ch Soft Error Counters 6  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Destination Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	00h
8	<b>Port 0 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
9	<b>Port 2 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
10	<b>Port 4 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
11	<b>Port 6 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
12	<b>Port 8 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
13	<b>Port 10 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
14	<b>Port 12 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
15	<b>Port 14 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
16	<b>Port 1 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
17	<b>Port 3 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
18	<b>Port 5 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
19	<b>Port 7 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
20	<b>Port 9 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
21	<b>Port 11 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
22	<b>Port 13 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
23	<b>Port 15 Source Queue Link List RAM 1-Bit Soft Error Counter Value</b>	RO	No	0
31:24	<b>Retry Buffer 1-Bit Soft Error Counter Value</b>	RO	No	00h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-474. F30h Soft Error Injection  
(Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Destination Queue Link List RAM Port A 1-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
1	<b>Destination Queue Link List RAM Port A 2-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
2	<b>Destination Queue Link List RAM Port A Error Injection Field Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RW	Yes	0
3	<b>Destination Queue Link List RAM Port B 1-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
4	<b>Destination Queue Link List RAM Port B 2-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
5	<b>Destination Queue Link List RAM Port B Error Injection Field Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RW	Yes	0
7:6	<b>Reserved</b>	RsvdP	No	00b
8	<b>Retry Buffer 1-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
9	<b>Retry Buffer 2-Bit Soft Error Injection Toggle</b> Every toggle injects an error.	RW	Yes	0
10	<b>Retry Buffer Soft Error Injection Field Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RW	Yes	0
31:11	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 14.16.18 Device-Specific Registers – Error Reporting (Offsets F40h – F4Ch)

This section details the Device-Specific Error Reporting registers, which shadow the corresponding **Device Control** register Error Reporting-related bits (offset 70h[2:0]), for each Port. **Table 14-49** defines the register map.

**Table 14-49. Device-Specific Error Reporting Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
	Error Reporting Enable for Ports 0, 1, 2, 3	F40h
	Error Reporting Enable for Ports 4, 5, 6, 7	F44h
	Error Reporting Enable for Ports 8, 9, 10, 11	F48h
	Error Reporting Enable for Ports 12, 13, 14, 15	F4Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-475. F40h Error Reporting Enable for Ports 0, 1, 2, 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 0 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 0.	RW	Yes	0
1	<b>Port 0 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 0.	RW	Yes	0
2	<b>Port 0 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 0.	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>Port 1 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 1.	RW	Yes	0
9	<b>Port 1 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 1.	RW	Yes	0
10	<b>Port 1 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 1.	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>Port 2 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 2.	RW	Yes	0
17	<b>Port 2 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 2.	RW	Yes	0
18	<b>Port 2 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 2.	RW	Yes	0
23:19	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>Port 3 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 3.	RW	Yes	0
25	<b>Port 3 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 3.	RW	Yes	0
26	<b>Port 3 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 3.	RW	Yes	0
31:27	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-476. F44h Error Reporting Enable for Ports 4, 5, 6, 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 4 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 4.	RW	Yes	0
1	<b>Port 4 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 4.	RW	Yes	0
2	<b>Port 4 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 4.	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>Port 5 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 5.	RW	Yes	0
9	<b>Port 5 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 5.	RW	Yes	0
10	<b>Port 5 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 5.	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>Port 6 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 6.	RW	Yes	0
17	<b>Port 6 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 6.	RW	Yes	0
18	<b>Port 6 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 6.	RW	Yes	0
23:19	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>Port 7 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 7.	RW	Yes	0
25	<b>Port 7 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 7.	RW	Yes	0
26	<b>Port 7 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 7.	RW	Yes	0
31:27	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-477. F48h Error Reporting Enable for Ports 8, 9, 10, 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 8 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 8.	RW	Yes	0
1	<b>Port 8 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 8.	RW	Yes	0
2	<b>Port 8 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 8.	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>Port 9 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 9.	RW	Yes	0
9	<b>Port 9 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 9.	RW	Yes	0
10	<b>Port 9 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 9.	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>Port 10 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 10.	RW	Yes	0
17	<b>Port 10 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 10.	RW	Yes	0
18	<b>Port 10 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 10.	RW	Yes	0
23:19	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>Port 11 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset 70h[0]), for Port 11.	RW	Yes	0
25	<b>Port 11 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset 70h[1]), for Port 11.	RW	Yes	0
26	<b>Port 11 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset 70h[2]), for Port 11.	RW	Yes	0
31:27	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-478. F4Ch Error Reporting Enable for Ports 12, 13, 14, 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Port 12 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 12.	RW	Yes	0
1	<b>Port 12 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 12.	RW	Yes	0
2	<b>Port 12 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 12.	RW	Yes	0
7:3	<b>Reserved</b>	RsvdP	No	0-0h
8	<b>Port 13 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 13.	RW	Yes	0
9	<b>Port 13 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 13.	RW	Yes	0
10	<b>Port 13 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 13.	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>Port 14 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 14.	RW	Yes	0
17	<b>Port 14 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 14.	RW	Yes	0
18	<b>Port 14 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 14.	RW	Yes	0
23:19	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>Port 15 Correctable Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Correctable Error Reporting Enable</i> bit (offset <i>70h[0]</i> ), for Port 15.	RW	Yes	0
25	<b>Port 15 Non-Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Non-Fatal Error Reporting Enable</i> bit (offset <i>70h[1]</i> ), for Port 15.	RW	Yes	0
26	<b>Port 15 Fatal Error Reporting Enable</b> Shadow copy of <b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <i>70h[2]</i> ), for Port 15.	RW	Yes	0
31:27	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 14.16.19 Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch)

This section details the Device-Specific ARI Capability registers, which shadow ARI-related bits at offsets 8Ch[5] and 90h[5], for each Port. Table 14-50 defines the register map.

**Table 14-50. Device-Specific ARI Capability Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
Device Control 2 for Port 0	Device Capability 2 for Port 0	F50h
Device Control 2 for Port 1	Device Capability 2 for Port 1	F54h
Device Control 2 for Port 2	Device Capability 2 for Port 2	F58h
Device Control 2 for Port 3	Device Capability 2 for Port 3	F5Ch
Device Control 2 for Port 4	Device Capability 2 for Port 4	F60h
Device Control 2 for Port 5	Device Capability 2 for Port 5	F64h
Device Control 2 for Port 6	Device Capability 2 for Port 6	F68h
Device Control 2 for Port 7	Device Capability 2 for Port 7	F6Ch
Device Control 2 for Port 8	Device Capability 2 for Port 8	F70h
Device Control 2 for Port 9	Device Capability 2 for Port 9	F74h
Device Control 2 for Port 10	Device Capability 2 for Port 10	F78h
Device Control 2 for Port 11	Device Capability 2 for Port 11	F7Ch
Device Control 2 for Port 12	Device Capability 2 for Port 12	F80h
Device Control 2 for Port 13	Device Capability 2 for Port 13	F84h
Device Control 2 for Port 14	Device Capability 2 for Port 14	F88h
Device Control 2 for Port 15	Device Capability 2 for Port 15	F8Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-479. F50h Device Capability and Control 2 for Port 0  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 0</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 0</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-480. F54h Device Capability and Control 2 for Port 1  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 1</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 1</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-481. F58h Device Capability and Control 2 for Port 2  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 2</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 2</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-482. F5Ch Device Capability and Control 2 for Port 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 3</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 3</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-483. F60h Device Capability and Control 2 for Port 4  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 4</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 4</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-484. F64h Device Capability and Control 2 for Port 5  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 5</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 5</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-485. F68h Device Capability and Control 2 for Port 6  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 6</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 6</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-486. F6Ch Device Capability and Control 2 for Port 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 7</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 7</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-487. F70h Device Capability and Control 2 for Port 8  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 8</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 8</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-488. F74h Device Capability and Control 2 for Port 9  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 9</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 9</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-489. F78h Device Capability and Control 2 for Port 10  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 10</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 10</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-490. F7Ch Device Capability and Control 2 for Port 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 11</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 11</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-491. F80h Device Capability and Control 2 for Port 12  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 12</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 12</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-492. F84h Device Capability and Control 2 for Port 13  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 13</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 13</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-493. F88h Device Capability and Control 2 for Port 14  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 14</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 14</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 14-494. F8Ch Device Capability and Control 2 for Port 15  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 15</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	RO	Yes	1
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 15</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 0 = Disables 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 14.17 Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

This section details the Advanced Error Reporting Extended Capability registers. Table 14-51 defines the register map.

**Table 14-51. 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)
	Uncorrectable Error Status	FB4h
	Uncorrectable Error Mask	FB8h
	Uncorrectable Error Severity	FCCh
	Correctable Error Status	FC0h
	Correctable Error Mask	FC4h
	Advanced Error Capabilities and Control	FC8h
	Header Log 0	FD0h
	Header Log 1	FD4h
	Header Log 2	FD8h
	Header Log 3	FDCh

**Register 14-495. 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>		RO	Yes	0001h
19:16	<b>Capability Version</b>		RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to 138h, which addresses the upstream Port/NT Port Link Interface <b>Power Budget Extended Capability</b> structure.	Upstream	RO	Yes	138h
	Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	Downstream	RO	Yes	148h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-496. FB8h Uncorrectable Error Status  
(All Ports)**

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>Uncorrectable Error Mask</b> register bits (offset FBCh).					
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> <i>Reserved/Not supported</i>		RO	No	1
14	<b>Completion Timeout Status</b> <i>Not applicable to switches.</i>		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
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 detected 1 = Violation is detected	Downstream	RW1CS <sup>a</sup>	Yes	0
31:22	<b>Reserved</b>		RsvdP	No	0-0h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (offset 1C8h[2]) is Set, Type changes from RW1CS to RW.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-497. FBCh Uncorrectable Error Mask  
(All Ports)**

Bit(s)	Description	Ports	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>Uncorrectable Error Status</b> register bits (offset <b>FB8h</b> ).					
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> <i>Reserved/Not supported</i>		RsvdP	No	0
14	<b>Completion Timeout Mask</b> <i>Not applicable to switches.</i>		RsvdP	No	0
15	<b>Completer Abort Mask</b>		RWS	Yes	0
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
31:22	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-498. FC0h Uncorrectable Error Severity  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
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	RO	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>		RO	No	1
14	<b>Completion Timeout Severity</b> <i>Not applicable to switches.</i> 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 reported as fatal		RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 14-498. FC0h Uncorrectable Error Severity  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	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> <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	Upstream Downstream	RsvdP RWS	No Yes	0 0
31:22	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 14-499. FC4h Correctable 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>Correctable Error Mask</b> register bits (offset FC8h).				
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
31:14	<b>Reserved</b>	RsvdP	No	0-0h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (offset 1C8h[2]) is Set,  
Type changes from RW1CS to RW.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-500. FC8h Correctable 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>Correctable Error Status</b> register bits (offset <b>FC4h</b> ).				
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
31:14	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 14-501. 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 <b>FB8h</b> ).	ROS	No	1Fh
5	<b>ECRC Generation Capable</b> 0 = ECRC generation is not supported 1 = ECRC generation is supported, but must be enabled	RO	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	RO	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

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 14-502. 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 14-503. 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 14-504. 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 14-505. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3



## Chapter 15 Non-Transparent Bridging – NT Mode Only

### 15.1 Introduction

*Note:* Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

The PEX 8619 supports Non-Transparent (NT) bridge functionality (NT mode), which is used to implement High-Availability systems or Intelligent I/O modules using PCI Express technology. The following discusses the basic NT bridging concept, as it applies to a PCI Express system.

NT bridges allow systems to isolate Address spaces, by appearing as an endpoint to the Host. The NT bridge exposes a Type 0 Configuration Space register (CSR) Header and forwards transactions from one domain to the other, using address translation. The NT bridge is used to connect two independent address/Host domains. The NT bridge includes **Doorbell** registers, for transmitting interrupts from one side of the bridge to the other. The bridge also includes **Scratchpad** registers, accessible from both domains for inter-Host communication. The PEX 8619, with a single Port configured to operate in NT mode, supports the **Intelligent Adapter Mode** system model. NT mode is enabled/disabled by the **STRAP\_NT\_ENABLE#** input.

The following are PEX 8619 Non-Transparent Bridging (NTB) key elements:

- Device Type Identification
- NT Port Features
- Intelligent Adapter Mode
- NT Port Reset
- NT Port Memory-Mapped Base Address Registers
- Doorbell Registers
- Scratchpad Registers
- NT Base Address Registers
- Address Translation

#### 15.1.1 Device Type Identification

Devices identify themselves by way of the Conventional PCI Configuration Space register (CSR) Header **PCI Class Code** register. A Transparent PCI-to-PCI bridge identifies itself as a **PCI Class Code 060400h**. An NT bridge identifies itself as “Other Bridge,” **068000h**, with a Type 0 Header, which is consistent with the use of other NT bridges available in the industry.

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 or downstream Port*, while an NT bridge/switch NT Port identifies itself as a *PCI Express endpoint*.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.1.2 NT Port Features

- Maps PEX 8619 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
- **Doorbell** registers
- **Scratchpad** registers
- Supports Requester ID and Completion ID translation
- NT Port Link Interface *DL\_Active* state change generates interrupt to Local 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 and NT Port for the enabling of High Availability systems (Failover and Redundant systems)
- Brings down the NT Port Link when the Local Host domain is down (if enabled)
- Supports the Fencing mechanism
- Signals Device-Specific interrupt to the Local Host when the NT Port Link Interface detects TLP errors
- Signals Device-Specific interrupt to the Local Host when the NT Port Link Interface receives Error Messages (Safety bit-controllable)
- Disables NT Port Link Interface Hot Reset effect (enabled, by default)
- Supports Configuration Space access control
- Option to appear behind a PCI-to-PCI bridge (refer to [STRAP\\_NT\\_P2P\\_EN#](#) ball description)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

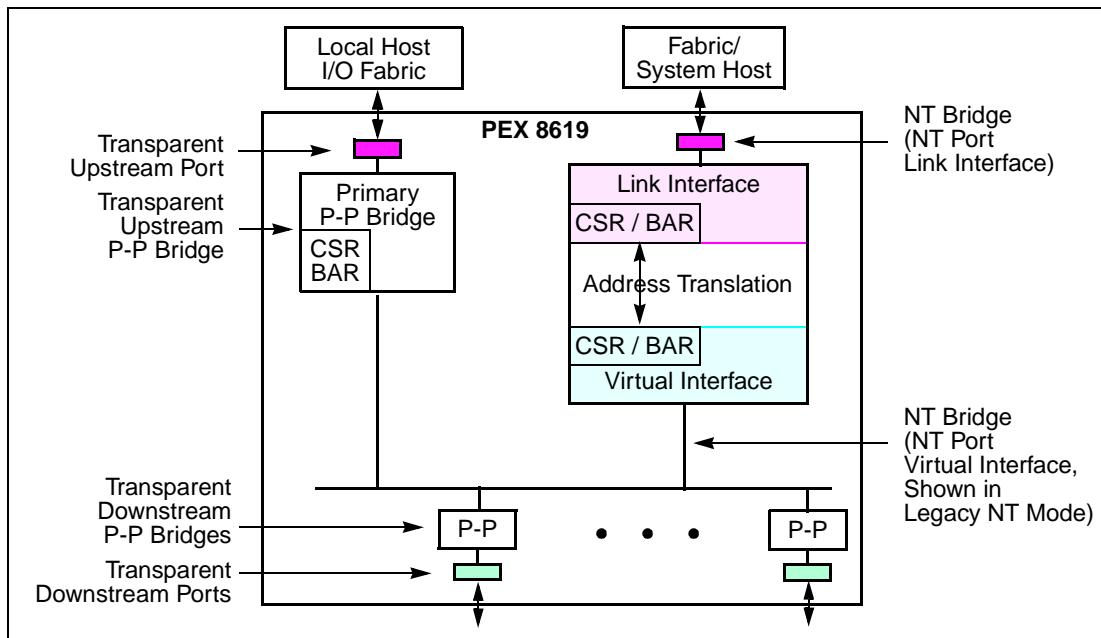
### 15.1.3 Intelligent Adapter Mode

The use of NT bridges in PCI systems is well-established for supporting intelligent adapters in enterprise and multi-Host systems. The same concept is used in PCI Express bridges and switches.

In [Figure 15-1](#), there are two Type 0 CSR Headers in the NT bridge. The one nearer the internal virtual PCI Bus is referred to as the *Virtual Interface*. The one nearer the PCI Express Link is referred to as the *Link Interface*.

In Intelligent Adapter mode, the NT Port Link Interface is connected to the System Host domain. The System Host manages only the NT Port Link Interface Type 0 function. The Local Host manages all PEX 8619 Transparent Port Type 1 and NT Port Virtual Interface Type 0 functions. Cross-domain traffic is routed through an Address Translation mechanism. (Refer to [Section 15.1.9](#).)

**Figure 15-1. Intelligent Adapter Software Model**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.1.4 NT Port Reset

The section discusses NT mode exceptions and enhancements to Transparent mode PCI Express (standard) reset behavior.

### 15.1.4.1 Fundamental Reset (PEX\_PERST#)

**PEX\_PERST#** resets all PEX 8619 states, including NT Port states. This reset initializes all Sticky bits and Configuration registers in Virtual and Link spaces to default values.

### 15.1.4.2 Intelligent Adapter Mode NT Port Reset

When the Transparent upstream Port receives a Hot Reset or enters the *DL\_Down* state, the PEX 8619, by default, propagates the in-band reset to all Transparent downstream Ports and connected downstream devices (to reset the downstream hierarchy), then resets the internal switch fabric and the NT Port Virtual Interface. There is no reset propagation to the NT Port Link Interface, and the Link-side remains intact.

When the NT Port Link Interface receives a Hot Reset or enters the *DL\_Down* state, the NT Port Link Interface registers are reset, by default. This Soft Reset does not reset the Transparent Ports nor the NT Port Virtual Interface. Instead, when the NT Port Link Interface receives a Hot Reset (or enters the *DL\_Down* state), the **PEX\_NT\_RESET#** output is asserted Low for 1 µs. The system can use this signal to trigger a reset of the entire Local subsystem.

The PEX 8619 supports an option that allows these Hot Reset conditions at its Transparent upstream Port and NT Port Link Interface to be masked (disabled) for all Ports, including the NT Port, by Setting the **Debug Control** register *Upstream Port and NT-Link DL\_Down Reset Propagation Disable* bit (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[20]).

When software writes to the PEX 8619 Transparent upstream Port's **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]), the resulting **Secondary Bus Reset** is (as above) propagated to all PEX 8619 Transparent downstream Ports, and the Port states and NT Port Virtual Interface states are reset.

### 15.1.5 NT Port Memory-Mapped Base Address Registers

The NT Port Virtual and Link Interfaces individually claim 128 KB of memory, using **BAR0** and **BAR1**. The 128-KB space contains the CSRs for all PEX 8619 Ports. **BAR0** and **BAR1** can be programmed as one of the following:

- 32-bit BAR (**BAR1** is *reserved*; default mode)
- 64-bit BAR, by programming the Configuration **BAR0/1 Setup** register (the NT Port Virtual Interface offset is **D0h**; the NT Port Link Interface offset is **E4h**)
- **BAR0** and **BAR1** can be completely disabled

Figure 15-2 provides a memory-mapped view of the PEX 8619 CSRs. This view is the same from the upstream Port, NT Port Virtual Interface, or NT Port Link Interface.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 15-2. NT Mode Configuration Register Mapping to Memory-Mapped BAR****PEX 8619**

Port 0	0 KB: 0_0000h
Port 1	4 KB: 0_1000h
Port 2	8 KB: 0_2000h
Port 3	12 KB: 0_3000h
Port 4	16 KB: 0_4000h
Port 5	20 KB: 0_5000h
Port 6	24 KB: 0_6000h
Port 7	28 KB: 0_7000h
Port 8	32 KB: 0_8000h
Port 9	36 KB: 0_9000h
Port 10	40 KB: 0_A000h
Port 11	44 KB: 0_B000h
Port 12	48 KB: 0_C000h
Port 13	52 KB: 0_D000h
Port 14	56 KB: 0_E000h
Port 15	60 KB: 0_F000h
NT Port Virtual Interface	64 KB: 1_0000h
NT Port Link Interface	68 KB: 1_1000h
<i>Reserved</i>	
	128 KB: 2_0000h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 15.1.6 Doorbell Registers

**Doorbell** registers are used to signal interrupts from one side of the NT bridge 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](#)
- [Virtual Interface IRQ Clear](#)
- [Virtual Interface IRQ Mask Set](#)
- [Virtual Interface IRQ Mask Clear](#)
- [Link Interface IRQ Set](#)
- [Link Interface IRQ Clear](#)
- [Link Interface IRQ Mask Set](#)
- [Link Interface IRQ Mask Clear](#)

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 and 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 and 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 MSIs or INTx. 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 Doorbell interrupts can optionally use the **PEX\_INTA#** output for interrupt signaling, instead of the INTx or MSI signaling mechanisms. PEX\_INTA# output can be enabled for NT Port Doorbell interrupts, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# Interrupt Output(s) for NT-Virtual Doorbell-Generated Interrupts* bit (NT Port Virtual Interface, or Port 0 if Port 0 is the NT Port, offset 1C8h[7]).

The PEX 8619 Virtual interrupts are de-asserted when the NT Port goes to the *DL\_Down* state.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.1.7 Scratchpad Registers

**Scratchpad** registers are readable and writable from both sides of the NT bridge, 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#**).

The NT bridge uses the Conventional PCI set of BARs in its Type 0 CSR Header to define Address ranges into the Memory space on the other side of the bridge. BARs define resource Address ranges that allow transaction forwarding to the opposite (other side) interface.

## 15.1.8 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 15.1.8.1](#).)
- **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, and/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 8619 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.)

### 15.1.8.1 NT BARx Setup Registers

PCI defines Base Address Registers (BARs) for a PCI device to claim Address space. For the PEX 8619, **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 15-1](#) 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 and/or firmware). *For example*, if the window size must be 1 MB, Memory space, and cacheable region, the **BARx Setup** register in 32-bit space will be FFF0\_0008h (FFF0\_0000h indicates the 1-MB space Request, bit 3 is the cacheable region, and bit 0 must be Memory space).

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).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 15-1. NT Port Virtual and Link Interface BARs**

BAR	NT Port Virtual Interface Description	NT Port Link Interface Description
<b>BAR0</b>	<p>All PEX 8619 Port Configuration registers are mapped into Memory space, using <b>BAR0</b> and <b>BAR1</b>. The Local Host, connected to the Transparent upstream Port, can use the Transparent upstream Port <b>BAR0/1</b> or NT Port Virtual Interface <b>BAR0/1</b> to access the PEX 8619 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 128-KB Memory space to the system.</p>	<p>All PEX 8619 Port Configuration registers are mapped into Memory space, using <b>BAR0</b> and <b>BAR1</b>. The System Host, connected to the NT Port, can use <b>BAR0/1</b> to access the PEX 8619 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 128-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 Virtual 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 (BAR2/3)</b> . <b>BAR2</b> uses <i>Direct Address Translation</i> .	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 (BAR2/3)</b> . <b>BAR2</b> uses <i>Direct Address Translation</i> .
<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 (BAR2/3)</b> . <b>BAR3</b> uses <i>Direct Address Translation</i> .	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 (BAR2/3)</b> . <b>BAR3</b> uses <i>Direct Address Translation</i> .
<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 (BAR4/5)</b> . <b>BAR4</b> uses <i>Direct Address Translation</i> .	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 (BAR4/5)</b> . <b>BAR4</b> uses <i>Direct Address Translation</i> .
<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 (BAR4/5)</b> . <b>BAR5</b> uses <i>Direct Address Translation</i> .	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 (BAR4/5)</b> . <b>BAR5</b> uses <i>Direct Address Translation</i> .

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.1.9 Address Translation

The Transparent bridge uses **Base** and **Limit** registers in I/O space, Non-Prefetchable Memory space, and Prefetchable Memory space to map transactions downstream, across the bridge. All downstream devices must be mapped in contiguous address regions, such that a single Address range in each space is sufficient. Upstream mapping is done by way of inverse decode, relative to the same registers. A Transparent bridge does not translate the addresses of forwarded transactions/packets.

In multi-domain systems, each Host domain has its own Address space, that is different 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 Requester ID, translations.

Before a transaction (PCI Express packet) can go through the NT bridge (either from the Virtual-side to Link-side, or from Link-side to Virtual-side) in an inter-domain system, one or more sets of Memory resources must be assigned to the NT bridge. To request this resource from the system enumerator (BIOS and/or firmware), the NT bridge must be programmed with the **BARx Setup** register(s). (Refer to [Section 15.1.8.1](#).) The **BARx Setup** register(s) requests the window space size, memory type, 32- or 64-bit space, 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 NT bridge in **BAR0** through **BAR5**. Any transactions that target **BAR2** through **BAR5** on the NT Port Link Interface result in a transaction across the NT bridge, to the secondary address domain.

Similarly, in NT PCI-to-PCI Bridge mode, the NT PCI-to-PCI bridge 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 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) can be assigned any available Bus Number within the upstream Port's range of Subordinate Bus Numbers.

In addition, the Lookup Table (LUT) register(s) and **Memory BARx Address Translation** register(s) must be programmed. The LUTs are the Requester ID (Bus Number, Device Number, and Function Number) with the ability to disable features that allow Requester's transaction go through the NT bridge (if enabled). This adds security to the NT bridge, limiting the devices that can generate transactions across the NT bridge. LUTs also play a crucial role, because the Requester ID is also used to complete PCI Express Read Requests – during the PCI Express Read Request to the other domain side, the NT bridge uses its own Requester ID to translate the original PCI Read Request, and when the Completion returns, the NT bridge uses the original Requester ID to complete the transaction.

The Address Translation is used to re-direct the address of the PCI Express packet to a programmer-reserved area (instead of using the same address for both Host domains). Hence, any transaction targeting **BAR2** through **BAR5** can be translated (re-mapped) on the other side of the NT bridge while maintaining the offsets. These Translation registers can be changed during runtime, as long as there are no pending transactions.

The PEX 8619 NT Port Virtual and Link Interfaces support [Direct Address Translation](#), described in the following section.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

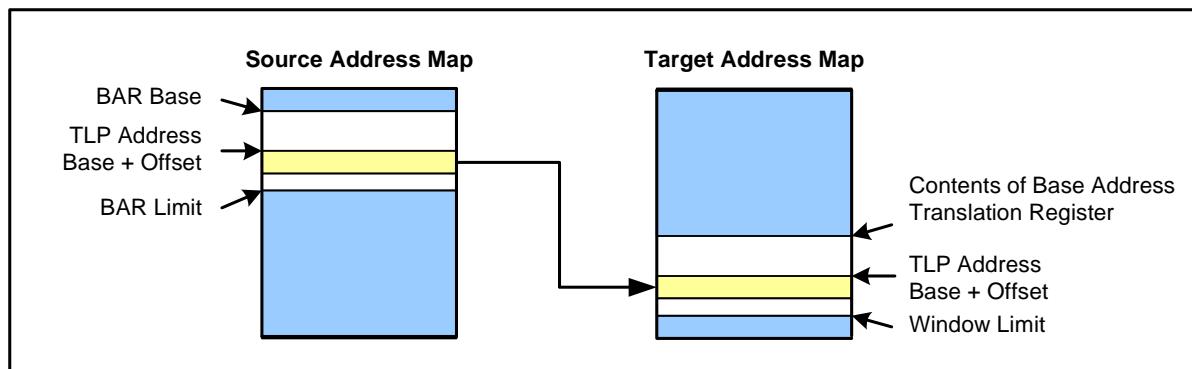
Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 15.1.9.1 Direct Address Translation

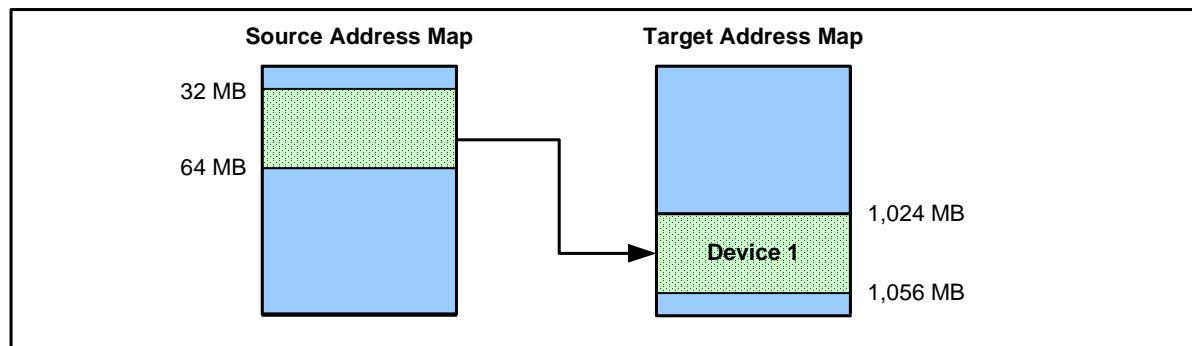
The **BARx Setup** registers define a mask that splits the address into an upper *Base* field and a lower *Offset* field. Translation then consists of replacing, under the maskable portion of the **BARx Setup** register, the Address Base register bits with the corresponding Address Translation register bits. Accordingly, the Address Translation register value must be a multiple of the corresponding BAR size. [Figure 15-3](#) 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 15-4](#) illustrates the entire Address map, claimed by the NT Port, mapped into the single target device. [Figure 15-5](#) illustrates the entire Address map claimed by the NT Port, mapped into multiple target devices. Multiple devices must be in contiguous Memory ranges.

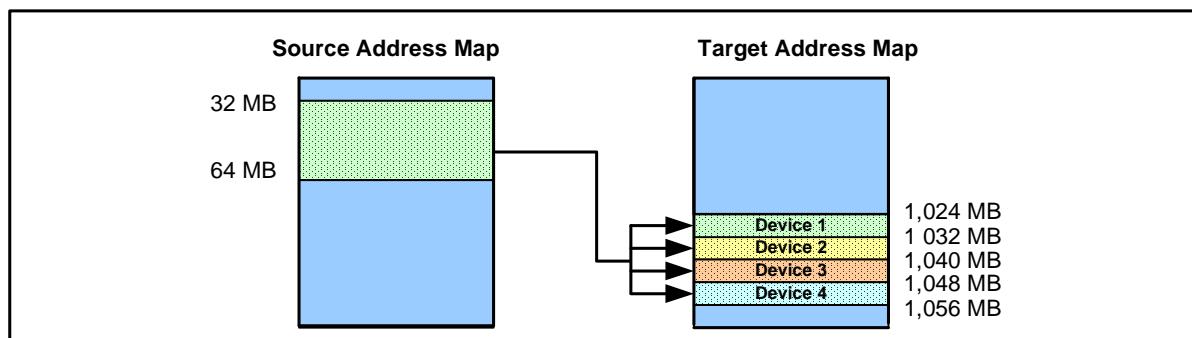
**Figure 15-3. Direct Address Translation**



**Figure 15-4. NT Port Mapped into Single Target Device**



**Figure 15-5. NT Port Mapped into Multiple Target Devices**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

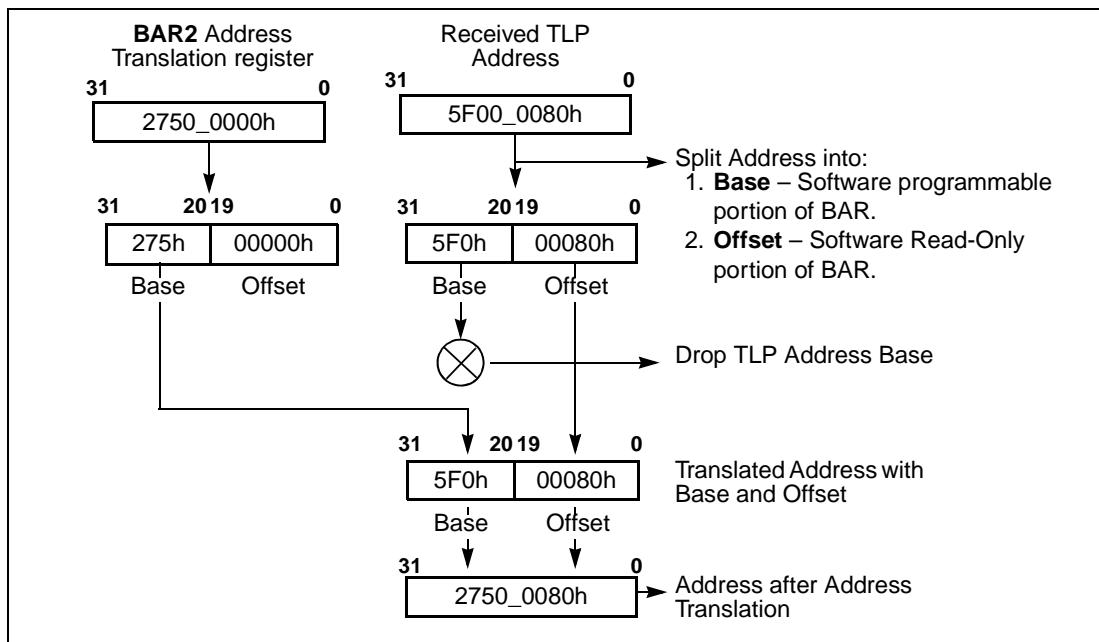
***Address Translation Example***

Assume the following:

1. NT Port Virtual Interface **BAR2** claims 128-KB Memory space (**BAR2 Setup** register = FFFE\_0000h).
2. Configuration software assigns the 5F00\_0000h address value to NT Port Virtual Interface **BAR2** and it is within the Transparent upstream Port Memory window.
3. Device driver software programs the **BAR2** Address Translation register to 2750\_0000h.

The PEX 8619 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 8619 claims the transaction and executes the address translation described in [Figure 15-6](#).

**Figure 15-6. Address Translation Example**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

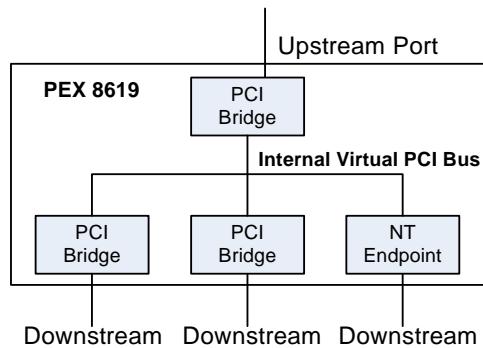
Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 15.2 NT PCI-to-PCI Bridge Mode

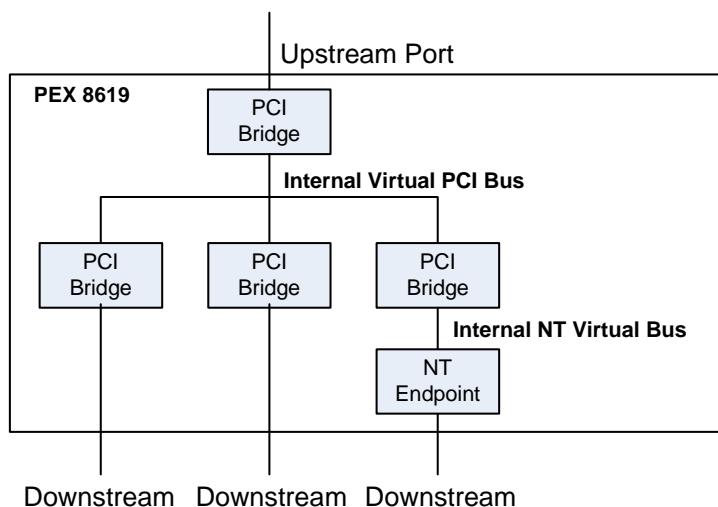
The PEX 8619 provides an option to insert a PCI-to-PCI bridge (NT PCI-to-PCI bridge) between the NT Port Virtual Interface and internal virtual PCI Bus. Without the NT PCI-to-PCI bridge, the NT Port is parallel to other Transparent downstream bridges within the device hierarchy. This mode is referred to as *Legacy NT mode*. (Refer to [Figure 15-7](#).) With the NT PCI-to-PCI bridge, the NT Port Virtual Interface is connected under one of the downstream bridges. (Refer to [Figure 15-8](#).)

The NT PCI-to-PCI bridge is not connected to a physical Link; therefore, there are some minor differences in the Configuration registers, particularly in some Link control functionality. Insertion of the NT PCI-to-PCI bridge does not affect packet latency, and is controlled by the **STRAP\_NT\_P2P\_EN#** input, when NT mode is enabled.

**Figure 15-7. Legacy NT Mode (STRAP\_NT\_P2P\_EN#=H)**



**Figure 15-8. NT PCI-to-PCI Bridge Mode (STRAP\_NT\_P2P\_EN#=L)**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.3 Requester ID Translation

Configuration, Message, and Completion transactions are ID-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 15-9 illustrates the Memory Request TLP Header format. Figure 15-10 illustrates the Completion TLP Header format.

**Figure 15-9. Memory Request TLP Header Format**

		Byte 0								Byte 1								Byte 2								Byte 3																					
Bytes 0-3		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														
		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 15-10. Completion TLP Header Format**

		Byte 0								Byte 1								Byte 2								Byte 3									
Bytes 0-3		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		
		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																

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 15.3.1 Transaction Sequence

*To implement a transaction sequence:*

1. Requester inserts ID information into the Memory Read TLP that it generates on the initiating Host domain.
2. Switches and bridges between the transaction initiator and PEX 8619 NT Port route this Memory Read TLP, based upon the address.
3. NT Port replaces the Memory Read TLP Requester ID with its ID, and conducts the address translation before it forwards this Requester ID-translated TLP to the target Host domain, because the NT Port is the transaction initiator in the target Host domain.
4. Switches and bridges between the PEX 8619 NT Port and target device route this Memory Read TLP, based upon the address.
5. When the target device generates the Completion TLP, it copies the Memory Read TLP Requester ID into the corresponding Completion TLP *Requester ID* field and inserts its ID into the TLP *Completer ID* field.
6. Switches and bridges between the target device and PEX 8619 NT Port route the Completion TLP, based upon Requester ID (in this case, NT Bridge ID) information.
7. NT Port restores the original Requester ID value from the Configuration register and implements another Requester ID and Completer ID translation for the Completion TLP, before it forwards the Completion TLP to the Requester Host domain.
8. Switches and bridges between the PEX 8619 NT Port and Requester route the Completion TLP, based upon the Requester ID (in this case, NT Bridge ID).
9. Requester accepts and processes the Completion TLP.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 15.3.2 Transaction Originating in Local Host Domain

The translation of outgoing Requests from the NT Port Virtual Interface to the NT Port Link Interface uses an 8-entry LUT, as discussed in [Section 16.14.2, “NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry \(Addresses D94h – DB0h\).”](#) Each LUT entry supports all outgoing Requests and any quantity of outstanding Requests made by a single device or function. If a device uses Phantom Function Numbers to increase the maximum quantity of outstanding transactions, each phantom function consumes an LUT entry. Configure the LUT by a serial EEPROM, I<sup>2</sup>C, and/or local firmware, so it is possible to transmit Requests to the System Host domain, which provides a measure of security and protection.

When a Memory Request arrives at the NT Port Virtual Interface, the packet Requester ID is associated with this LUT. If it attains one of the enabled LUT entries, the corresponding entry address (TxIndex) is inserted into the *Function Number* field of the packet’s Requester ID. Conversely, if it does not match one of the enabled LUT entries, an Unsupported Request (UR) Completion is returned.

At the same time, the contents of the NT Port Link Interface **Bus Number** and **Device Number Capture** registers (the values used during the last CSR Write to the Port) are copied into the packet Requester ID’s *Bus Number* and *Device Number* fields.

A Completion, with translated Requester ID, returned from the System Host domain to the PEX 8619, is recognized when its Requester ID Bus Number and Device Number match the NT Port Link Interface captured Bus Number and Device Number. (Refer to [Figure 15-11](#).)

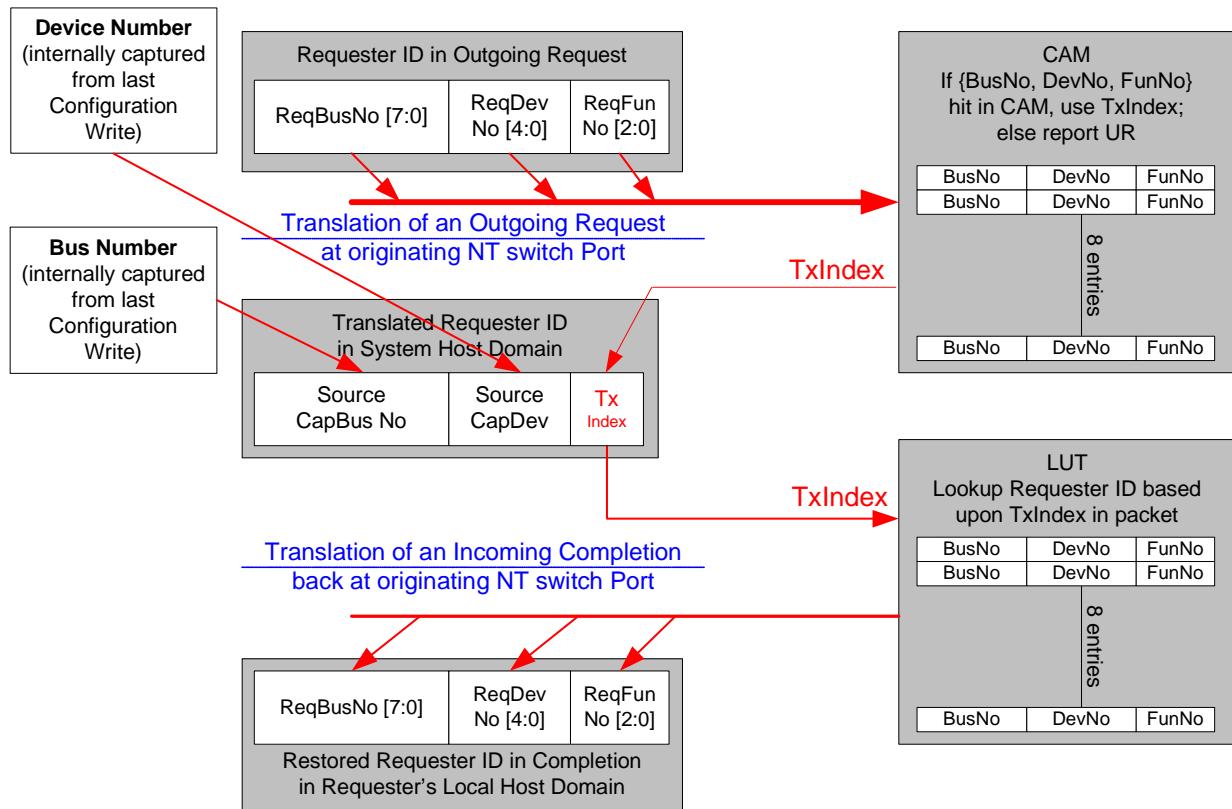
When the original Requester ID is restored, the following occurs:

1. TxIndex is retrieved from the *Function Number* field of the Completion TLP Requester ID.
2. TxIndex is used to look up the same 8-entry LUT, to restore the original Requester ID.
3. If the selected entry is valid, the restored Requester ID is placed into the Completion *TLP Requester* field; otherwise, an Unexpected Completion is reported.
4. Completion TLP *Completer ID* field is replaced by the NT Port Virtual Interface captured Bus Number, Device Number, and Function Number.
5. Translated Completion TLP is forwarded to the original Requester, in the Local Host domain.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Figure 15-11. Requester ID Translation for Request Originating in Local Host Domain**

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 15.3.3 Transaction Originating in System Host Domain

Transactions originating in the System Host domain use a Receive LUT, with 32 entries, as discussed in Section 17.15.1, “NT Port Link Interface Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h).” This data structure supports up to 32 devices (elsewhere in the System Host domain) that are transmitting Requests through the associated NT Port. Because the Function Number is not used in the LUT association, a separate LUT entry is not required for each requesting or phantom function device. Configure the LUT 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 from the System Host domain and routed to the NT Port, its Requester ID is translated again – Bus Number and Device Number, but not Function Number. The received Memory Request TLP Requester ID is associated with this LUT, and the address (RxIndex) of the corresponding matching entry is substituted into the *Device Number* field of the Memory Request’s TLP *Requester ID* field.

If no match is found, or the matched entry is not enabled, the Request receives a UR response. If a match is found, and matched entry is enabled, the PEX 8619 internal virtual PCI Bus Number is copied into the packet Requester ID’s *Bus Number* field. The translated Memory Request TLP is address-translated and forwarded into the Local Host domain.

The PEX 8619 internal virtual PCI 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 the switch internal virtual PCI Bus. Elsewhere within the PCI Express hierarchy, the Bus Number is sufficient to route the Completion back into the switch containing the NT Port.

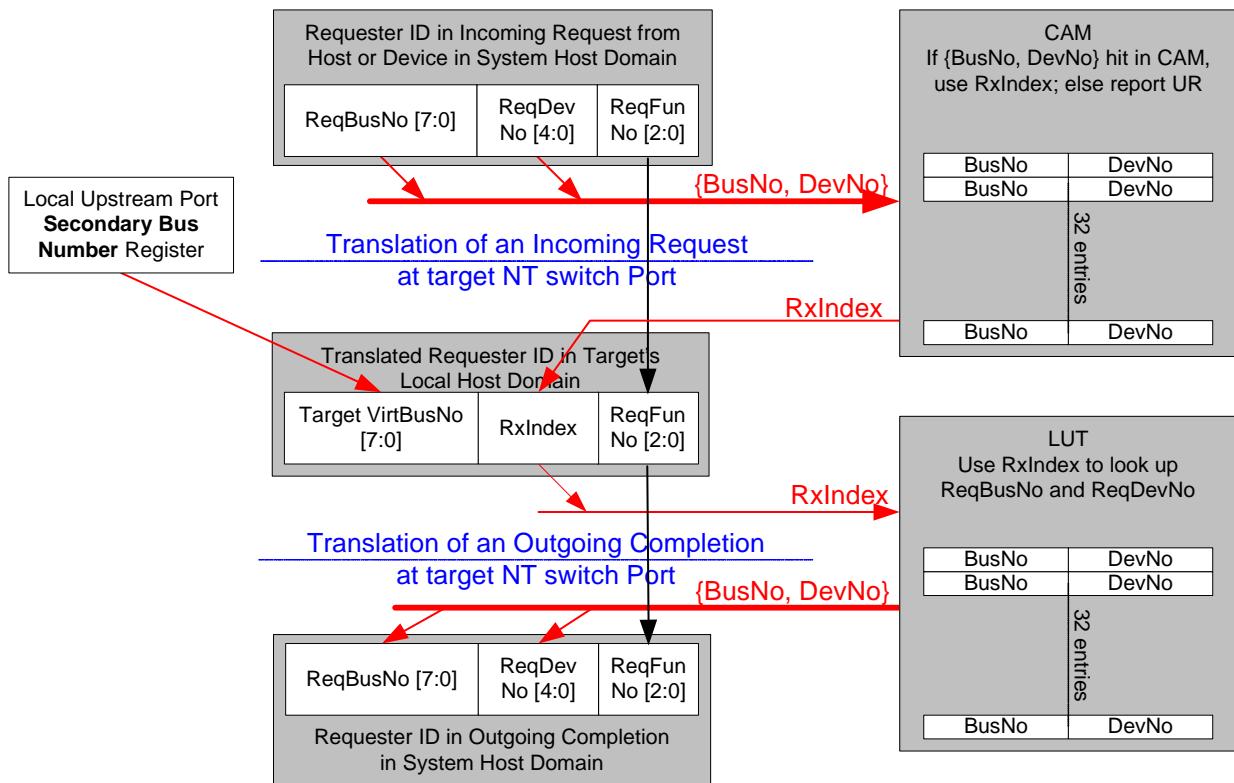
The inverse translation occurs when a Completion passes through the NT bridge from the Local Host domain to the System Host domain. The RxIndex is retrieved from the *Device Number* field of the received Completion TLP *Requester ID* Header field, and used to look up the 32-entry LUT. The Completion TLP *Requester ID*, *Bus Number*, and *Device Number* fields are replaced by the decoded 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 Link Interface captured Bus Number, Device Number, and Function Number values before forwarding the Completion TLP to the System Host domain. (Refer to [Figure 15-12](#).)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Figure 15-12. Requester ID Translation for Request Originating in System Host Domain**

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.4 Traffic Class Translation

For Memory Requests originating from the NT Port, the Traffic Class (TC) Translation registers can be programmed to translate the original Traffic Class of the TLP to a higher-priority Traffic Class (TC[7-1]). This TC translation allows use for both VC resources available in the PEX 8619, for traffic differentiation and Quality of Service (QoS). Most Root Complexes implement a single Virtual Channel (VC). With a single VC implementation, there is no differentiation of traffic, because all Requests and Completions are queued, in order, in the same VC resources. Moreover, typically, Root Complexes support a single Traffic Class, TC0, when generating Requests.

The NT Port can be used to provide a separate path to the CPU. However, because the Root Complex can support only TC0, the TC translation allows the PEX 8619 to modify the Traffic Class for incoming TLPs, thus providing traffic differentiation.

For Posted TLPs (Memory Writes) received at the NT Port Link Interface, the PEX 8619 modifies the current TC in the packet Header, with the programmed TC. The updated TLP can make use of the multiple Virtual Channels available in the PEX 8619.

For Non-Posted TLPs (Memory Reads), the PEX 8619 changes the TC of the Memory Read Request so that it can take advantage of any additional VC resources. When the Target responds with a Completion, the PEX 8619 changes the Completion TC of the Completion back to the original TC (TC0), before the TC is forwarded out from the NT Port Link Interface. The TC Translation registers are located in the NT Port Link Interface, at register offset [960h](#).

## 15.5 NT Port Power Management Handling

### 15.5.1 Active State Power Management

The NT Port Link Interface endpoint supports the Active State Power Management (ASPM) L0s and L1 Link PM states. The NT Port Virtual Interface endpoint implements the CSRs for ASPM support. However, it does not enter into the low-power states, because there is no physical Link associated with it.

The PEX 8619 Transparent Ports also support the ASPM L0s and L1 Link PM states in Transparent mode.

### 15.5.2 PCI-PM and PME Turn Off Support

When NT mode is enabled, the NT Port Link Interface Type 0 endpoint behaves as any other endpoints in the PCI Express PCI-PM D3hot Device PM state. Once in the D3hot Device PM state, the NT Port Link Interface Type 0 endpoint Requests PCI-PM L1 Link PM state entry and finally settles in the L1 Link PM state. Only Configuration accesses and Messages to the NT Port Link Interface Type 0 endpoint are supported in the D3hot Device PM state. The Root Complex transmits PME\_Turn\_Off Messages when the NT Host decides to turn Off the main power and Reference Clock. The PEX 8619 NT Port Link Interface Type 0 endpoint indicates its readiness to lose power, by transmitting a PME\_TO\_Ack Message toward 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 PEX 8619 Transparent upstream Port, the Port broadcasts this Message to all PEX 8619 downstream devices, including the NT Port Virtual Interface Type 0 endpoint. After the PME\_TO\_Ack Message is received from all downstream devices and the PEX 8619 NT Port Virtual Interface Type 0 endpoint, the PEX 8619 Transparent upstream Port transmits 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 PEX 8619 Transparent downstream Ports allow the attached devices to enter the PCI-PM-compatible L1 Link PM state. The PEX 8619 NT Port Virtual Interface Type 0 endpoint never enters the PCI-PM L1 Link PM state.

### 15.5.3 Message Generation

The PEX 8619 NT Port Link Interface Type 0 endpoint never generates PM\_PME Messages. The PEX 8619 NT Port Virtual Interface Type 0 endpoint never receives Set\_Slot\_Power\_Limit Messages nor generates PM\_PME Messages.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.6 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 Control** register *Expansion ROM Virtual Side* bit (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 660h[23]).

The NT Port supports 16- or 32-KB-sized Expansion ROM, based upon the **Serial EEPROM Clock Frequency** register *Expansion ROM Size* bit (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 268h[16]) value.

*Note:* Expansion ROM can be enabled in either the NT Port Virtual Interface or NT Port Link Interface, but not both simultaneously. Expansion ROM is enabled, by default, in the NT Port Link Interface (*Ingress Control* register *Expansion ROM Virtual Side* bit is Cleared).

## 15.7 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 INTx Message Requests. If the NT Port receives an INTx Message Request, the Request is handled as a Malformed TLP error.

### 15.7.1 NT Port Virtual Interface Interrupts

The NT Port Virtual Interface generates interrupts to the Local 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 INTx, MSI, or **PEX\_INTA#** signaling mechanisms (all mutually exclusive), as follows:

- PEX\_INTA# output can be enabled for NT-Virtual Doorbell interrupts, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# Interrupt Output(s) for NT-Virtual Doorbell-Generated Interrupts* bit (NT Port Virtual Interface, or Port 0 if Port 0 is the NT Port, offset 1C8h[7]).
- PEX\_INTA# output can be enabled for NT-Link Error and Event interrupts, by Setting the register's *Enable PEX\_INTA# Interrupt Output(s) for Device-Specific Error-Triggered Interrupts* bit (offset 1C8h[5])

Refer to [Section 15.1.6](#) for Doorbell interrupt details.

The NT Port Virtual Interface de-asserts INTx or PEX\_INTA# interrupts in response to one or more of the following conditions:

- NT Port Virtual Interface **PCI Command** register *Interrupt Disable* bit (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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

In NT PCI-to-PCI Bridge mode, 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 Local 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 as Port 0, 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 Local 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 as Port 0, 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 Local Host, if the interrupt signaling 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 as Port 0, 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 Local 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 as Port 0, offsets **FE0h[3]** and **FE4h[3]**), respectively).

## 15.7.2

## NT Port Link Interface Interrupts

The NT Port Link Interface generates interrupts to the System 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.

NT-Link Doorbell interrupts can use the INTx, MSI, or **PEX\_INTA#** signaling mechanisms (all mutually exclusive). PEX\_INTA# output can be enabled for NT-Link Doorbell interrupts, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# Interrupt Output(s) for NT-Link Doorbell-Generated Interrupts* bit (NT Port Link Interface, offset **1C8h[7]**). Refer to [Section 15.1.6](#) for Doorbell interrupt details.

The NT Port Link Interface de-asserts INTx or PEX\_INTA# interrupts in response to one or more of the following conditions:

- NT Port Link Interface **PCI Command** register *Interrupt Disable* bit (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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.8 NT Port Error Handling

The PEX 8619 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 8619 signals Error Messages to the Local Host (Host closest to the upstream Port). The PEX 8619 provides an option to communicate this error condition to the System Host (Host closest to the NT Port), by signaling an interrupt.

The PEX 8619 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 8619 signals Error Messages to the System Host (Host closest to the NT Port).

When the PEX 8619 receives a TLP, it performs the following:

1. TLP integrity check,
2. Address decode,
3. Address translation,
4. Requester ID translation, and
5. ECRC re-generation,

before transmitting the TLP through the NT Port. If the PEX 8619 detects an ECRC error, it corrupts the re-generated ECRC before transmitting the TLP through the NT Port. The PEX 8619 also provides options for dropping error-detected ECRC or endpoint (EP) TLPs (**Ingress Control Shadow** register *Drop ECRC TLPs* and *Drop EP TLPs* bits (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [664h\[8, 4\]](#), respectively)).

The PEX 8619 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 8619 drops all TLPs traveling from the NT Port Virtual Interface to the NT Port Link Interface, if the internal RAM Fatal ECC error is detected, until the PEX 8619 receives a Hot Reset from a Local Host.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 15.8.1 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 Control Shadow** register *NT Error Message Drop* bit (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **664h[11]**) is Set, the NT Port Link Interface drops the Message, and logs the error in the **Link Error Status Virtual** register *Link Side Uncorrectable Error Message Drop Status* bit (NT Port Virtual Interface as Port 0, offset **FE0h[3]**). If the corresponding **Link Error Mask Virtual** register *Link Side Uncorrectable Error Message Drop Mask* bit (NT Port Virtual Interface as Port 0, offset **FE4h[3]**) is Set, the NT Port Virtual signals an interrupt (INTx, MSI, or **PEX\_INTA#**) to the Local Host through the upstream Port, if interrupts are enabled.

### 15.8.2 NT PCI-to-PCI Bridge Mode Error Handling

In NT PCI-to-PCI Bridge mode, 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, 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 PCI-to-PCI Bridge 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**)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 15.9 Cursor Mechanism

A software application can use the **Device-Specific Cursor Mechanism** to access the PEX 8619 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 16.10, “NT Port Virtual Interface Vendor-Specific Capability 3 Registers \(Offsets C8h – FCh\)”](#)
- [Section 17.10, “NT Port Link Interface Vendor-Specific Capability 3 Registers \(Offsets C8h – FCh\)”](#)

## 15.10 Port Programmability

The PEX 8619 supports the capability of programming the upstream Port and NT Port Number. The Configuration register for the upstream Port and NT Port is in the **Debug Control** register *Upstream Port ID* and *NT Port Number* fields (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 1DCh[11:8, 27:24], respectively). This register is updated, based upon the external **STRAP\_NT\_ENABLE#** and **STRAP\_NT\_UPSTRM\_PORTSEL[3:0]** inputs, by default. A serial EEPROM, I<sup>2</sup>C, and/or software can be used to override the external strap values.

A software application can change the upstream Port and NT Port location to another Port Number during runtime, or as part of a failover sequence. It is recommended that the PEX 8619 be in an Idle state (no traffic) when changing the 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.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 16 NT Port Virtual Interface Registers – NT Mode Only

### 16.1 Introduction

*Note:* Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

The NT Port includes two sets of Configuration, Capability, Control, and Status registers, to support the Virtual and Link Interfaces. This chapter defines the NT Port Virtual Interface registers. Other registers are defined in:

- Chapter 14, “Transparent Port Registers”
- Chapter 17, “NT Port Link Interface Registers – NT Mode Only”
- Chapter 18, “DMA Interface Registers – DMA Mode Only”

*Note:* For Chip-specific registers (those that exist only in Port 0), if Port 0 is a Legacy NT Port, then those registers exist only in the NT Port Virtual Interface.

All PEX 8619 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 r2.0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

855

## 16.2 NT Port Virtual Interface Type 0 Register Map

Table 16-1 defines the NT Port Virtual Interface Type 0 register mapping.

**Table 16-1. 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	
		00h
		...
NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)	Capability Pointer (40h)	34h
		...
	Next Capability Pointer (48h)      Capability ID (01h)	3Ch
NT Port Virtual Interface PCI Power Management Capability Registers (Offsets 40h – 44h)		40h
	Next Capability Pointer (68h)      Capability ID (05h)	44h
NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h)		48h
	Next Capability Pointer (A4h)      Capability ID (10h)	...
NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h)		64h
	Next Capability Pointer (C8h)      SSID/SSVID Capability ID (0Dh)	68h
NT Port Virtual Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)		...
	Next Capability Pointer 3 (00h)      Capability ID 3 (09h)	A0h
NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)		...
Next Capability Offset (FB4h)	1h	PCI Express Extended Capability ID (0003h)
		100h
NT Port Virtual Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)		...
		134h
	<i>Reserved</i>	138h –
Next Capability Offset (448h or 950h)	1h	PCI Express Extended Capability ID (0002h)
		144h
NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)		...
		148h
		...
		1BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 16-1. NT Port Virtual Interface Type 0 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	
		NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h)
Next Capability Offset (950h)	1h	PCI Express Extended Capability ID (000Bh)
		NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h)
Next Capability Offset 2 (C34h)	1h	PCI Express Extended Capability ID 2 (000Bh)
		NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h)
Next Capability Offset 4 (000h)	1h	PCI Express Extended Capability ID 4 (000Bh)
		NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h)
		NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – EECh)
		NT Port Virtual Interface Device-Specific Registers (Offsets E40h – F8Ch)
	<i>Reserved</i>	
		F90h –
Next Capability Offset (148h)	1h	PCI Express Extended Capability ID (0001h)
		NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)
		NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.3 Register Access

The PEX 8619 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 8619 supports three mechanisms for accessing the NT Port Virtual Interface registers:

- PCI Express Base r2.0 Configuration Mechanism
- Device-Specific Memory-Mapped Configuration Mechanism
- Device-Specific Cursor Mechanism

### 16.3.1 PCI Express Base r2.0 Configuration Mechanism

The *PCI Express Base r2.0* 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.

#### 16.3.1.1 PCI r3.0-Compatible Configuration Mechanism

The *PCI r3.0*-Compatible Configuration mechanism provides standard access to the PEX 8619 NT Port Virtual Interface's first 256 bytes (the bytes at offsets 00h through FFh) of the PCI Express Configuration Space. (Refer to [Figure 16-1](#).)

The mechanism uses PCI Type 0 and Type 1 Configuration transactions to access the PEX 8619 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 8619 upstream Port, as required by the *PCI Express Base r2.0*.

The PEX 8619 decodes all Type 1 Configuration accesses received on its upstream Port, when any of the following conditions exist:

- If the Bus Number specified in the Configuration access is the Bus Number of the PEX 8619 internal virtual PCI Bus, the PEX 8619 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 8619 Transparent downstream Ports), the PEX 8619 processes the Read or Write Request
  - If the specified Device Number does not correspond to any of the PEX 8619 downstream Port Device Numbers nor NT Port Number, the PEX 8619 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).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

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

#### 16.3.1.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism 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.

This mechanism can be used to access all PEX 8619 Configuration registers.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 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 128-KB Memory map, as illustrated in [Figure 16-1](#). This Memory map is identical for Upstream Port **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. In NT PCI-to-PCI Bridge mode (**STRAP\_NT\_P2P\_EN#=L**), 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 **Debug Control** register *NT Port Number* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[27:24]**)).

To use this mechanism, use the [PCI r3.0-Compatible Configuration Mechanism](#) to program the PEX 8619 NT Port Virtual Interface **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, offsets **10h** and **14h**, respectively). After the PEX 8619 NT Port Virtual Interface Memory-Mapped register Base address is Set, the PEX 8619 Configuration Space registers are accessed, using Memory Reads and Writes to the 4-KB range, starting at offset 64 KB (1\_0000h, Virtual Interface) and offset 68 KB (1\_1000h, Link Interface).

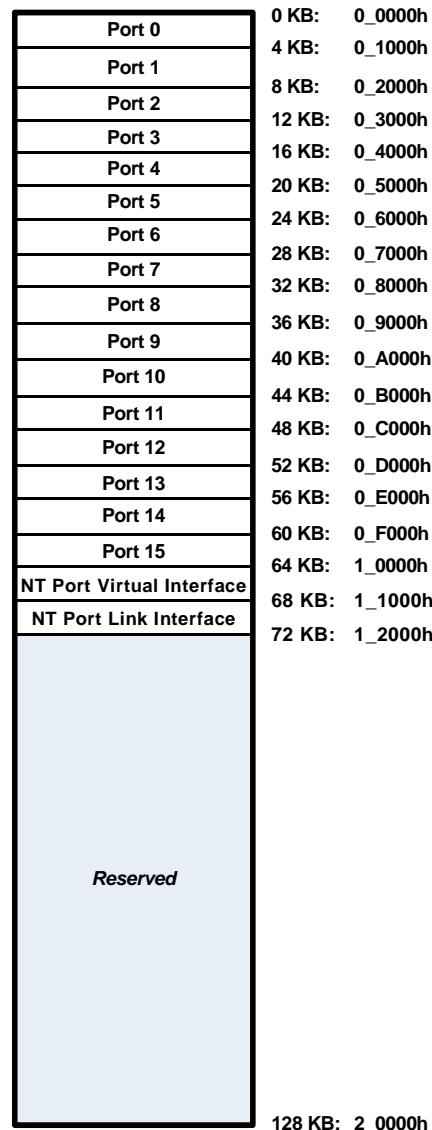
NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Figure 16-1. NT Mode Configuration Register Mapping to Memory-Mapped BAR**

PEX 8619

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

### 16.3.3 Device-Specific Cursor Mechanism

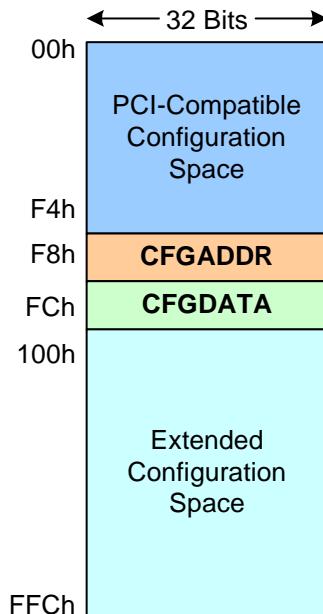
The Device-Specific Cursor mechanism is provided for use in development systems that can only generate *PCI r3.0* Configuration cycles (*that is*, the system cannot use either the [Device-Specific Memory-Mapped Configuration Mechanism](#), nor generate Extended Configuration Requests to access the Extended Configuration Space).

In [Figure 16-2](#), the software uses the **Configuration Address Window** (CFGADDR) register (offset F8h) to point to the NT Port Virtual or Link Interface Configuration Space registers, including the PCI Express Extended Configuration Space registers (offsets 100h through FFCh).

Software uses the **Configuration Data Window** (CFGDATA) register (offset FCh) to write to or read from the selected Configuration Space registers.

Refer to [Section 16.10, “NT Port Virtual 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 8619 NT Port Virtual Interface registers, including:

- Bit/field names
- Description of register functions in the PEX 8619 NT Port Virtual and Link Interfaces
- Type (*such as* RW or HwInit; refer to [Table 14-5, “Register Types, Grouped by User Accessibility,”](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8619 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 16.5 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 16-2](#) defines the register map.

**Table 16-2. 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
		Base Address 0	0Ch
		Base Address 1	10h
		Base Address 2	14h
		Base Address 3	18h
		Base Address 4	1Ch
		Base Address 5	20h
		<i>Reserved</i>	24h
Subsystem ID		Subsystem Vendor ID	28h
		Expansion ROM Base Address	2Ch
	<i>Reserved</i>		30h
			34h
		<i>Reserved</i>	38h
Max_Lat ( <i>Reserved</i> )	Min_Gnt ( <i>Reserved</i> )	PCI Interrupt Pin	3Ch
		PCI Interrupt Line	

### 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 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**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 PEX 8619 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 8619 ignores Memory Space Requests on the NT Port Virtual Interface 1 = PEX 8619 accepts Memory Space Requests received on the NT Port Virtual Interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 forwarding of Memory Requests upstream. Does not affect Message forwarding nor Completions. 0 = PEX 8619 handles Memory Requests received on the NT Port Link Interface as Unsupported Requests (UR); for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 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 r2.0</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 r2.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</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 r2.0</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-2. 04h PCI Command/Status (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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– <b>PEX_INTA#</b> (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 r2.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</i> .	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## Register 16-2. 04h PCI Command/Status (Cont.)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<p><b>Master Data Parity Error Detected</b></p> <p>If bit 6 (<i>Parity Error Response Enable</i>) is Set, the NT Port Virtual Interface Sets this bit when the NT Port:</p> <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> <p>If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8619 never Sets this bit.</p> <p>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.</p>	RW1C	Yes	0
26:25	<p><b>DEVSEL# Timing</b> <i>Not supported</i></p>	RsvdP	No	00b
27	<p><b>Signaled Target Abort</b></p> <p>The NT Port Virtual Interface Sets this bit if any of the following conditions exist:</p> <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 8619 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 8619 register address within a non-existent Port</li> <li>NT Port Virtual Interface receives a Memory Write Request targeting enabled Expansion ROM Address space (Expansion ROM Base Address Register (BAR), offset 30h)</li> </ul> <p><i>Note:</i> 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.</p>	RW1C	Yes	0
28	<p><b>Received Target Abort</b> <i>Reserved</i></p>	RsvdP	No	0
29	<p><b>Received Master Abort</b> <i>Reserved</i></p>	RsvdP	No	0
30	<p><b>Signaled System Error</b></p> <p>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.</p>	RW1C	Yes	0
31	<p><b>Detected Parity Error</b></p> <p>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.</p> <p>1 = NT Port Virtual Interface received a Poisoned TLP, regardless of the bit 6 (<i>Parity Error Response Enable</i>) state</p>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**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 8619. The PEX 8619 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>				
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 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 8619 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 r2.0</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 -- kotlin-novator | vtuz vova

**Register 16-5. 10h Base Address 0**  
**(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> 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), 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 D0h[1:0]) to 11b (which enables both <b>BAR0</b> and <b>BAR1</b> ).				
	When software writes to the NT Port Virtual Interface <b>BAR0</b> , the value is automatically copied to the NT Port Virtual Interface <b>BAR0 (Shadow Copy)</b> register (NT Port Virtual Interface, offset D68h).			
0	<b>Memory Space Indicator</b> When enabled, the Base Address register maps PEX 8619 Port Configuration registers into Memory space.  <i>Note:</i> 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8619 Port Configuration registers into Non-Prefetchable Memory space	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	0-0h
31:17	<b>Base Address 0</b> 128-KB-aligned Base address used for Memory-Mapped access to the 128-KB block of all PEX 8619 registers (4 KB, per Port).	RW	Yes	0-0h

**Register 16-6. 14h Base Address 1**  
**(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> When software writes to the NT Port Virtual Interface <b>BAR1</b> , the value is automatically copied to the NT Port Virtual Interface <b>BAR1 (Shadow Copy)</b> register (NT Port Virtual Interface, offset D6Ch).				
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 10h[2:1]) is programmed to 10b.  RO when the <b>Base Address 0</b> register ( <b>BAR0</b> ) is not enabled as a 64-bit BAR ( <b>Memory Map Type</b> field (offset 10h[2:1]) is not programmed to 10b).	RW	Yes	0000_0000h
		RO	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-7. 18h Base Address 2 (**  
**NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> When software writes to the NT Port Virtual Interface <b>Base Address 2</b> register ( <b>BAR2</b> ), the value is automatically copied to the NT Port Virtual Interface <b>BAR2 (Shadow Copy)</b> register (NT Port Virtual Interface, offset <b>D70h</b> ).				
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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	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 Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> When software writes to the NT Port Virtual Interface <b>Base Address 3</b> register ( <b>BAR3</b> ), the value is automatically copied to the NT Port Virtual Interface <b>BAR3 (Shadow Copy)</b> register (NT Port Virtual Interface, offset <b>D74h</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 <b>18h[2:1]</b> ), is programmed to 10b).				
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 space only supported	Offset 18h[2:1]=00b Offset 18h[2:1]=10b	RsvdP RW	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 All other encodings are <i>reserved</i> .	Offset 18h[2:1]=00b Offset 18h[2:1]=10b	RsvdP RW	00b 00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 18h[2:1]=00b Offset 18h[2:1]=10b	RsvdP RW	No 0
19:4	<b>Reserved</b> 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]=00b Offset 18h[2:1]=10b	RsvdP RW	0_000h 0_000h
31:20	<b>Base Address 3</b>	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-9. 20h Base Address 4**  
**(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> When software writes to the NT Port Virtual Interface <b>Base Address 4</b> register ( <b>BAR4</b> ), the value is automatically copied to the NT Port Virtual Interface <b>BAR4 (Shadow Copy)</b> register (NT Port Virtual Interface, offset <a href="#">D78h</a> ).				
0	<b>Memory Space Indicator</b> 0 = Memory space only 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	0_000h
31:20	<b>Base Address 4</b>	RW	Yes	000h

**Register 16-10. 24h Base Address 5**  
**(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> When software writes to the NT Port Virtual Interface <b>Base Address 5</b> register ( <b>BAR5</b> ), the value is automatically copied to the NT Port Virtual Interface <b>BAR5 (Shadow Copy)</b> register (NT Port Virtual Interface, offset <a href="#">D7Ch</a> ).				
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 <a href="#">20h[2:1]</a> ), is programmed to 10b).				
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 space only supported	Offset 20h[2:1]=00b Offset 20h[2:1]=10b	RsvdP RW	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 All other encodings are <i>reserved</i> .	Offset 20h[2:1]=00b Offset 20h[2:1]=10b	RsvdP RW	00b 00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 20h[2:1]=00b Offset 20h[2:1]=10b	RsvdP RW	0 0
19:4	<b>Reserved</b> 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]=00b Offset 20h[2:1]=10b	RsvdP RW	0_000h 0_000h
31:20	<b>Base Address 5</b>	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-11. 2Ch Subsystem ID and Subsystem Vendor ID**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

**Register 16-12. 30h Expansion ROM Base Address**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> 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 ( <a href="#">Ingress Control</a> register <a href="#">Expansion ROM Virtual Side</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 660h[23]) is Cleared).				
0	<b>Expansion ROM Enable</b> 0 = NT Port Virtual Interface Expansion ROM is disabled	NT Port Link Interface, offset 30h[0]=1	RsvdP	No 0
	1 = NT Port Virtual Interface Expansion ROM is enabled, and NT Port Link Interface Expansion ROM is disabled	NT Port Link Interface, offset 30h[0]=0	RW	Yes 0
		NT Port Link Interface, offset 30h[0]=1	RsvdP	No 0
13:1	<b>Reserved</b>	RsvdP	No	0-0h
31:14	<b>Expansion ROM Base Address</b> If the <a href="#">Serial EEPROM Clock Frequency</a> register <a href="#">Expansion ROM Size</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 268h[16]) value is 0, the Expansion ROM size is 16 KB (default value is FFFF_C001h). Bit 14 is RW. If the <a href="#">Expansion ROM Size</a> bit value is 1, the Expansion ROM size is 32 KB (default value is FFFF_8001h). Bit 14 is RO.	When Bit 0 = 0	RsvdP	No 0-0h
	When Bit 0 = 1	RW	Yes 0-0h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**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
<b>PCI Interrupt Line</b>				
7:0	<b>Interrupt Line Routing Value</b> 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
<b>PCI Interrupt Pin</b>				
15:8	<b>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 8619. 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>Min_Gnt</b>				
23:16	<b>Minimum Grant</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h
<b>Max_Lat</b>				
31:24	<b>Maximum Latency</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 16.6 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 16-3](#) defines the register map.

**Table 16-3. 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	PCI Power Management Data	Next Capability Pointer (48h)	Capability ID (01h)	40h
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 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 8619 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 8619 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8619 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. <i>Note:</i> This field is <i>reserved</i> in NT PCI-to-PCI Bridge mode.	RO	Yes	0000_0b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**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	<p><b>Power State</b>            Used to determine the current Device PM state of the Port, and to Set the Port into a new Device PM state.</p> <p>00b = D0            01b = D1 – <i>Not supported</i>            10b = D2 – <i>Not supported</i>            11b = D3hot</p> <p>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.</p>	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<p><b>No Soft Reset</b>            1 = Devices transitioning from the D3hot to D0 Device PM state, because of Power State commands, do not perform an internal reset</p>	RO	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<p><b>PME Enable</b>            Tied to 0, because the PEX 8619 does <i>not</i> generate PME in PCI Express mode.</p> <p><i>Note:</i> This bit is reserved in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0
12:9	<p><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 <b>Data</b> and <b>Data Scale</b> registers (fields [31:24] and 14:13, respectively).</p> <p>0h = D0 power consumed            3h = D3hot power consumed            4h = D0 power dissipated            7h = D3hot power dissipated</p> <p>All other encodings are reserved.</p>	RO	Yes	0h
14:13	<p><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>Data</b> register value. The value and meaning of this field varies, depending upon which data value is selected by field [12:9] (<i>Data Select</i>). There are four internal <b>Data Scale</b> registers (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h.</p>	RO	Yes	00b
15	<p><b>PME Status</b>            0 = PME is not being generated by the NT Port</p> <p><i>Note:</i> This bit is reserved in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**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	<b>Reserved</b>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <b>Reserved</b> 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 <b>Data Scale</b> value returned is 0h. Selected by field [12:9] ( <i>Data Select</i> ).	RO	Yes	00h

- a. With no serial EEPROM nor previous I<sup>2</sup>C programming, Reads return 00h for the **Data** and **Data Scale** registers (for all *Data Selects*).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.7 NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h)

The registers detailed in [Section 14.8, “MSI Capability Registers \(Offsets 48h – 64h\),”](#) are also applicable to the NT Port Virtual Interface. [Table 16-4](#) defines the register map used by the NT Port Virtual Interface.

**Table 16-4. 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
<i>Reserved</i>		MSI Data		54h
	MSI Mask			58h
	MSI Status			5Ch
	<i>Reserved</i>			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.

## 16.8 NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h)

The registers detailed in [Section 14.9, “PCI Express Capability Registers \(Offsets 68h – A0h\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-5](#) (register map; offsets 7Ch and 80h are *reserved*), and [Register 16-17](#) through [Register 16-22](#).

**Table 16-5. 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 (A4h)	Capability ID (10h) 68h
Device Status	<i>Not Supported/Reserved</i>	Device Control 70h
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	
<i>Reserved</i>		90h
Link Status 2	Link Control 2	
<i>Reserved</i>		94h
		98h
		9Ch – A0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-17. 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 r2.0</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 8619 NT Port Virtual Interface programs this field to 2h, as required by the <i>PCI Express Base r2.0</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> <i>Note:</i> This bit is also reserved in NT PCI-to-PCI Bridge mode.	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-18. 6Ch Device Capability**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
2:0	<p><b>Maximum Payload Size Supported</b>            Maximum Payload Size Port limitations are as follows:            If there are no x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 2,048 bytes. If there are x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 512 bytes.</p> <p>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.</p>	HwInit	Yes	100b
4:3	<p><b>Phantom Functions Supported</b>  <i>Not supported</i></p>	RO	Yes	00b
5	<p><b>Extended Tag Field Supported</b>            0 = Maximum Tag field is 5 bits            1 = Maximum Tag field is 8 bits</p>	RO	Yes	0
8:6	<p><b>Endpoint L0s Acceptable Latency</b>            111b = No Limit</p>	RO	Yes	111b
11:9	<p><b>Endpoint L1 Acceptable Latency</b>            111b = No Limit</p>	RO	Yes	111b
14:12	<b>Reserved</b>	RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>	RO	Yes	1
17:16	<b>Reserved</b>	RsvdP	No	00b
25:18	<p><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>).</p>	RO	Yes	00h
27:26	<p><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</p>	RO	Yes	00b
31:28	<b>Reserved</b>	RsvdP	No	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-19. 70h Device Status and Control**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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 Local 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 Local 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 Local 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 Local Host	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> <i>Not supported</i>	RsvdP	No	0
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 <b>Maximum Payload Size Supported</b> 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.  <i>Note:</i> Software must halt all transactions through the NT Port before changing this field.	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>	RsvdP	No	0
14:12	<b>Maximum Read Request Size</b> <i>Not supported</i>	RsvdP	No	000b
15	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-19. 70h Device Status and Control (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> Set when the NT Port Virtual Interface detects a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state. 0 = NT Port Virtual Interface did not detect a Correctable error 1 = NT Port Virtual Interface detected a Correctable error	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> Set when the NT Port Virtual Interface detects a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state. 0 = NT Port Virtual Interface did not detect a Non-Fatal error 1 = NT Port Virtual Interface detected a Non-Fatal error	RW1C	Yes	0
18	<b>Fatal Error Detected</b> Set when the NT Port Virtual Interface detects a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state. 0 = NT Port Virtual Interface did not detect a Fatal error 1 = NT Port Virtual Interface detected a Fatal error	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> Set when the NT Port Virtual Interface detects a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state. 0 = NT Port Virtual Interface did not detect a UR 1 = NT Port Virtual Interface detected a UR	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 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-20. 74h Link Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Supported Link Speeds</b>            Indicates the NT Port Virtual Interface's supported Link speed.            0001b = 2.5 GT/s Link speed is supported            0010b = 5.0 GT/s and 2.5 GT/s Link speeds are supported            All other encodings are <i>reserved</i>.</p>	RO	Yes	0010b (STRAP_RESERVED17#=H) 0001b (STRAP_RESERVED17#=L)
9:4	<p><b>Maximum Link Width</b>            The PEX 8619 maximum Link width is x8 = 00_1000b. Valid widths are x1, x4, or x8. Actual maximum Link width is Set by the <b>STRAP_PORTCFG[3:0]</b> inputs.            00_0000b = <b>Reserved</b>            00_0001b = x1            00_0010b = x2            00_0100b = x4            00_1000b = x8            All other encodings are <i>not supported</i>.</p>	ROS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>574h[3:0]</b> )
11:10	<p><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>.</p>	RO	Yes	11b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-20. 74h Link Capability (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
14:12	<p><b>L0s Exit Latency</b>            Indicates the L0s Link PM state exit latency for the given PCI Express Link. Value depends upon the <b>Common Physical Layer Command/Status</b> register <i>N_FTS Value</i> (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 238h[15:8]) value and Link speed. Exit latency is calculated, as follows:           <ul style="list-style-type: none"> <li><b>2.5 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 <i>N_FTS</i>) x 4 ns (1 symbol time at 2.5 GT/s)</li> <li><b>5.0 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 <i>N_FTS</i>) x 2 ns (1 symbol time at 5.0 GT/s)</li> </ul>           100b = NT Port Virtual Interface L0s Link PM state Exit Latency is 512 ns to less than 1 µs at 5.0 GT/s            101b = NT Port Virtual Interface L0s Link PM state Exit Latency is 1 µs to less than 2 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.  <i>Note:</i> The NT Port Virtual Interface never enters the L0s Link PM state, because there is no physical Link attached to it.</p>	RO	No	100b (5.0 GT/s)  101b (2.5 GT/s)
17:15	<p><b>L1 Exit Latency</b>            Indicates the L1 Link PM state exit latency for the given PCI Express Link. Value depends upon the Link speed.            001b = NT Port Virtual Interface L1 Link PM state Exit Latency is 1 µs to less than 2 µs at 5.0 GT/s            010b = NT Port Virtual Interface L1 Link PM state Exit NT Latency is 2 µs to less than 4 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.  <i>Note:</i> The NT Port Virtual Interface never enters the L1 Link PM state, because there is no physical Link attached to it.</p>	RO	Yes	001b (5.0 GT/s)  010b (2.5 GT/s)
18	<b>Clock Power Management</b>	RO	Yes	0
23:19	<p><b>Reserved</b>  <i>Note:</i> Bits [21:19] are also <i>reserved</i> in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-20. 74h Link Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:24	<p><b>Port Number</b>  The NT Port Number value is selected by the <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs.  All other encodings are <b>reserved</b>.</p> <p>00h = Port 0  01h = Port 1  02h = Port 2  03h = Port 3  04h = Port 4  05h = Port 5  06h = Port 6  07h = Port 7  08h = Port 8  09h = Port 9  0Ah = Port 10  0Bh = Port 11  0Ch = Port 12  0Dh = Port 13  0Eh = Port 14  0Fh = Port 15</p>	ROS	No	Set by STRAP_NT_UPSTRM_PORTSEL[3:0] input levels

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-21. 78h Link Status and Control**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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> <i>Reserved</i> for the NT Port Virtual Interface.	RsvdP	No	0
5	<b>Retrain Link</b> <i>Reserved</i> 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-21. 78h Link Status and Control (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>Link Status</b>				
19:16	<p><b>Current Link Speed</b>            Indicates the negotiated Link speed of the Port's PCI Express Link.            0001b = 2.5 GT/s Link speed            0010b = 5.0 GT/s Link speed            All other encodings are <i>reserved</i>. The value in this field is undefined when the Link is not up.</p>	RO	No	0001b
25:20	<p><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.            00_0000b = Link is down (default)            00_0001b = x1 or Port is in the <i>DL_Down</i> state            00_0010b = x2            00_0100b = x4            00_1000b = x8            All other encodings are <i>not supported</i>.</p>	RO	No	00_0000b
26	<p><b>Training Error</b>  <b>Reserved</b> for the NT Port Virtual Interface.</p>	RsvdP	No	0
27	<p><b>Link Training</b>  <b>Reserved</b> for the NT Port Virtual Interface. Always read as 0.</p>	RsvdP	No	0
28	<p><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 8619 uses an independent clock.</p>	HwInit	Yes	0
31:29	<b>Reserved</b>	RsvdP	No	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-22. 98h Link Status and Control 2**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>Link Control 2</b>				
3:0	<b>Target Link Speed</b> 0001b = 2.5 GT/s Link speed supported 0010b = 5.0 GT/s Link speed supported All other encodings are <i>reserved</i> .	RWS	Yes	0010b
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> 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
12	<b>Compliance De-Emphasis</b> Sets the de-emphasis level in the <i>Polling.Compliance</i> state, if the entry occurred due to bit 4 ( <a href="#">Enter Compliance</a> ) being Set.	RWS	Yes	0
15:13	<b>Reserved</b>	RsvdP	No	000b
<b>Link Status 2</b>				
16	<b>Current De-Emphasis Level</b> Reflects the de-emphasis level. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at 2.5 GT/s)	RO	Yes	0 (5.0 GT/s) 1 (2.5 GT/s)
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.9 NT Port Virtual Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)

The registers detailed in [Section 14.10, “Subsystem ID and Subsystem Vendor ID Capability Registers \(Offsets A4h – FCh\),”](#) are also applicable to the NT Port, except as defined in [Table 16-6](#) (register map) and [Register 16-23](#).

**Table 16-6. NT Port Virtual Interface Subsystem ID and Subsystem Vendor ID 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		
<i>Reserved</i>	Next Capability Pointer (C8h)	SSID/SSVID Capability ID (0Dh)	A4h
Subsystem ID	Subsystem Vendor ID		A8h
	<i>Reserved</i>		ACh – C4h

**Register 16-23. A4h Subsystem Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>SSID/SSVID Capability ID</b> Detects the 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> Program to C8h, to point to the <b>Vendor-Specific Capability 3</b> structure.	RO	Yes	C8h
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 16.10 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 16-7](#) defines the register map used by the NT Port.

The **NT Port Virtual Interface BARx Setup** (offsets D0h through E0h) register values are shadowed in the corresponding **NT Port Virtual Interface BARx Setup Shadow** registers (offsets D80h through D90h, respectively). When software writes to an **NT Port Virtual Interface BARx Setup** register, the value is automatically copied to the corresponding **NT Port Virtual Interface BARx Setup (Shadow Copy)** register. If the **NT Port Virtual Interface BARx Setup** registers are programmed by serial EEPROM, the **NT Port Virtual Interface BARx Setup (Shadow Copy)** registers must also be programmed by serial EEPROM, to the same respective register values.

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 r2.0*, and not the Device-Specific registers. However, if Port 0 is a Legacy NT Port, the Cursor Mechanism in the NT Port Virtual Interface registers can also access the Device-Specific registers.

**Table 16-7. 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		
Vendor-Specific Capability 3	Next Capability Pointer 3 (00h)	Capability ID 3 (09h)	C8h
Vendor-Specific Header 3 ( <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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-24. 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>	RO	Yes	09h
15:8	<b>Next Capability Pointer 3</b> 00h = This capability is the last capability in the Linked List	RO	Yes	00h
31:16	<b>Length</b> Quantity of bytes in this Capability structure.	RO	Yes	0038h

**Register 16-25. CCh Vendor-Specific Header 3**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Reserved</b>	RO	Yes	0380_0002h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-26. D0h NT Port Virtual Interface BAR0/1 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The NT Port Virtual Interface <b>BAR0/1 Setup (Shadow Copy)</b> register (offset D80h) must be programmed with the same value as this register.				
1:0	<b>BAR0/1 Enable</b> 00b = Disables Virtual Interface <b>BAR0</b> and <b>BAR1</b> 01b = <b>Reserved</b> 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)	RW	Yes	10b
2	<b>BAR0 Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
31:3	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-27. D4h NT Port Virtual Interface Memory BAR2 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The NT Port Virtual Interface <b>Memory BAR2 Setup (Shadow Copy)</b> register (offset D84h) must be programmed with the same value as this register. This requirement applies only to the NT Port Virtual Interface.				
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.	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	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 0, and Writes are ignored 1 = Corresponding <b>BAR2</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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] ( <b>BAR2 Type</b> ) = 00b	RW	Yes
	<b>BAR2 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <b>BAR2 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-28. D8h NT Port Virtual Interface Memory BAR2/3 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The NT Port Virtual Interface <b>Memory BAR2/3 Setup (Shadow Copy)</b> register (offset D88h) must be programmed with the same value as this register. This requirement applies only to the NT Port Virtual Interface.				
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.	RO	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	Offset D4h[2:1] ( <b>BAR2 Type</b> ) = 00b	RsvdP	No
	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] ( <b>BAR2 Type</b> ) = 10b	RWS	Yes
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 0, and Writes are ignored 1 = Corresponding <b>BAR3</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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] ( <b>BAR2 Type</b> ) = 00b	RW	Yes
	<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] ( <b>BAR2 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-29. DCh NT Port Virtual Interface Memory BAR4 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The NT Port Virtual Interface <b>Memory BAR4 Setup (Shadow Copy)</b> register (offset D8Ch) must be programmed with the same value as this register. This requirement applies only to the NT Port Virtual Interface.				
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.	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	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 0, and Writes are ignored 1 = Corresponding <b>BAR4</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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] ( <b>BAR4 Type</b> ) = 00b	RW	Yes
	<b>BAR4 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <b>BAR4 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-30. E0h NT Port Virtual Interface Memory BAR4/5 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The NT Port Virtual Interface <b>Memory BAR4/5 Setup (Shadow Copy)</b> register (offset D90h) must be programmed with the same value as this register. This requirement applies only to the NT Port Virtual Interface.				
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.	RO	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	Offset DCh[2:1] ( <b>BAR4 Type</b> ) = 00b	RsvdP	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] ( <b>BAR4 Type</b> ) = 10b	RWS	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 0, and Writes are ignored 1 = Corresponding <b>BAR5</b> bits are RW bits <i>Note:</i> If BAR[30:n] is the Base field (BAR size is 2 <sup>n</sup> ), BAR[30:n] should have all ones (1).	RW	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] ( <b>BAR4 Type</b> ) = 00b	RW	Yes
	<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] ( <b>BAR4 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-31. F8h Configuration Address Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Reserved</b>	RsvdP	No	0000h
25:16	<b>Register Offset</b>	RW	Yes	0-0h
30:26	<b>Reserved</b>	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 16-32. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.11 NT Port Virtual Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)

The registers detailed in [Section 14.11, “Device Serial Number Extended Capability Registers \(Offsets 100h – 134h\),”](#) are also applicable to the NT Port. [Table 16-8](#) defines the register map used by all Ports.

**Table 16-8. NT Port Virtual Interface Device Serial Number 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 (FB4h)	Capability Version (1h)	PCI Express Extended Capability ID (0003h)	100h
	Serial Number (Lower DW)		104h
	Serial Number (Upper DW)		108h
	<i>Reserved</i>		10Ch – 134h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 16.12 NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)

The registers detailed in [Section 14.13, “Virtual Channel Extended Capability Registers \(Offsets 148h – 1BCh\);”](#) are also applicable to the NT Port Virtual Interface, except for the Next Capability Offset value, as defined in [Table 16-9](#) (register map), and [Register 16-33](#) through [Register 16-35](#).

**Table 16-9. 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 (448h or 950h)	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
	VC1 Resource Capability	164h
	VC1 Resource Control	168h
VC1 Resource Status	<i>Reserved</i>	16Ch
	<i>Reserved</i>	170h –
NT Port Virtual Interface WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh) (Legacy NT Mode)		1A4h
<i>Reserved</i> (NT PCI-to-PCI Bridge Mode)		1A8h
...		
		1BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-33. 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 r2.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</i> .	RO	No	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>Vendor-Specific Extended Capability</b> structure, offset 448h.	NT Port is Port 0	RO	No
	Next extended capability is the <b>Vendor-Specific Extended Capability 2</b> structure, offset 950h.	Otherwise	RO	No
				448h
				950h

**Register 16-34. 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> 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]) 1 = Port Arbitration Table entry was written to by software	RO	No	0
17	<b>VC0 Negotiation Pending</b> 0 = VC0 negotiation is complete 1 = VC0 initialization is not complete for the NT Port Virtual Interface	RO	Yes	0
31:18	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-35. 16Ch VC1 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 supported</i>	RO	No	0
17	<b>VC1 Negotiation Pending</b> 0 = VC1 negotiation is complete 1 = VC1 initialization or disabling is pending for the NT Port Virtual Interface	RO	Yes	0
31:18	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 16.12.1 NT Port Virtual Interface WRR Port Arbitration Table Registers (Offsets 1A8h – 1BCh)

The registers detailed in [Section 14.13.1, “WRR Port Arbitration Table Registers \(Offsets 1A8h – 1BCh\),”](#) are also applicable to the NT Port Virtual Interface if the NT Port is the upstream Port (refer to the Note). Port Arbitration Table phases are used to determine Port weighting during “Weighted Round-Robin with 32 Phases” Port arbitration.

[Table 16-10](#) defines the register map used by the NT Port Virtual Interface. The numbers along the top of the table indicate the 4-bit fields of each 32-bit register. There are 32 phases, and any active Port Number can go into each Port  $x$  Phase  $x$  box.

**Notes:** *The Port Arbitration Table is used only when Weighted Round-Robin (WRR) with 32-phase Port Arbitration is selected. The Arbiter is parked on the upstream Port, if the upstream Port is Port 0 through Port 3. Moreover, the fields within this register are valid only on the upstream Port. If the upstream Port is on a higher-numbered Port, the Arbiter is parked on Port 0. Furthermore, the fields within this register are valid only on Port 0 and reserved on all other Ports.*

*This register structure is reserved in NT PCI-to-PCI Bridge mode.*

**Table 16-10. NT Port Virtual Interface WRR Port Arbitration Table Register Map (When WRR is enabled)**

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 $x$ Phase 7	Port $x$ Phase 6	Port $x$ Phase 5	Port $x$ Phase 4	Port $x$ Phase 3	Port $x$ Phase 2	Port $x$ Phase 1	Port $x$ Phase 0																							1A8h	
Port $x$ Phase 15	Port $x$ Phase 14	Port $x$ Phase 13	Port $x$ Phase 12	Port $x$ Phase 11	Port $x$ Phase 10	Port $x$ Phase 9	Port $x$ Phase 8																					1ACh			
Port $x$ Phase 23	Port $x$ Phase 22	Port $x$ Phase 21	Port $x$ Phase 20	Port $x$ Phase 19	Port $x$ Phase 18	Port $x$ Phase 17	Port $x$ Phase 16																					1B0h			
Port $x$ Phase 31	Port $x$ Phase 30	Port $x$ Phase 29	Port $x$ Phase 28	Port $x$ Phase 27	Port $x$ Phase 26	Port $x$ Phase 25	Port $x$ Phase 24																					1B4h			
<i>Reserved</i>																													1B8h – 1BCh		

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.13 NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – C88h)

The registers detailed in [Section 14.14, “Device-Specific Registers \(Offsets 1C0h – 51Ch\),”](#) and [Section 14.16, “Device-Specific Registers \(Offsets 530h – F8Ch\)”](#) (for offsets 558h through C88h), are unique to the PEX 8619 and not referenced in the *PCI Express Base r2.0*. These registers are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-11 \(register map\)](#) through [Table 16-15](#), and [Register 16-36](#) through [Register 16-65](#).

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- [Section 16.15, “NT Port Virtual Interface Device-Specific Registers \(Offsets E40h – F8Ch\)”](#)
- [Section 16.17, “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 16-11. NT Port Virtual Interface Device-Specific Register Map (Offsets 1C0h – 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	
		1C0h
NT Port Virtual Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)		...
		1FCh
		200h
Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)		...
		25Ch
Device-Specific Registers – Serial EEPROM (Offsets 260h – 26Ch) (Legacy NT Mode)		260h
<b>Reserved</b> (NT PCI-to-PCI Bridge Mode)		...
		26Ch
<b>Factory Test Only/Reserved</b>	270h –	288h
Device-Specific Registers – Miscellaneous Control (Offset 28Ch)		28Ch
		290h
Device-Specific Registers – I <sup>2</sup> C Slave Interface (Offsets 290h – 2C4h)		...
		2C4h
		2C8h
Device-Specific Registers – Bus Number CAM (Offsets 2C8h – 304h)		...
		304h
		308h
Device-Specific Registers – I/O CAM (Offsets 308h – 340h)		...
		340h
Device-Specific Registers – SMBus Slave Interface (Offset 344h)		344h
		348h
Device-Specific Registers – Address-Mapping CAM (Offsets 348h – 444h)		...
		444h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 16-11. NT Port Virtual Interface Device-Specific Register Map  
(Offsets 1C0h – C88h) (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 (950h)	1h	PCI Express Extended Capability ID (000Bh)
Device-Specific Registers – Vendor-Specific Dual Cast Extended Capability (Offsets 448h – 51Ch)		448h ... 51Ch
<b>Reserved</b>		520h – 540h
Device-Specific Registers – Read Pacing (Offsets 544h – 554h) (Legacy NT Mode) <i>Reserved</i> (NT PCI-to-PCI Bridge Mode)		544h ... 554h
<b>Factory Test Only</b>		558h – 570h
Device-Specific Registers – Port Configuration (Offsets 574h – 628h)		574h ... 628h
Device-Specific Registers – General-Purpose Input/Output (Offsets 62Ch – 65Ch) (Legacy NT Mode) <i>Reserved</i> (NT PCI-to-PCI Bridge Mode)		62Ch ... 65Ch
NT Port Virtual Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)		660h ... 67Ch
Device-Specific Registers – IOCAM Base and Limit Upper 16 Bits (Offsets 680h – 6BCh)		680h ... 6BCh
Device-Specific Registers – Base Address Shadow (Offsets 6C0h – 73Ch)		6C0h ... 73Ch
Device-Specific Registers – Virtual Channel Resource Control Shadow, DMA BAR, and DMA Read and Write Latency (Offsets 740h – 83Ch)		740h ... 83Ch
<b>Reserved</b>		840h – 93Ch
Device-Specific Registers – Ingress Credit Handler Port Pool (Offsets 940h – 94Ch)		940h ... 94Ch
Next Capability Offset 2 (C34h)	1h	PCI Express Extended Capability ID 2 (000Bh)
NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)		950h ... 95Ch
<b>Factory Test Only</b>		960h – 974h
<b>Reserved</b>		978h – 9FCh

**Table 16-11. NT Port Virtual Interface Device-Specific Register Map  
(Offsets 1C0h – C88h) (Cont.)**

### 16.13.1 NT Port Virtual Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)

The registers detailed in [Section 14.14.1, “Device-Specific Registers – Error Checking and Debug \(Offsets 1C0h – 1FCh\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-12](#) (register map; offset 1E4h is *reserved*), and [Register 16-36](#) through [Register 16-40](#).

**Table 16-12. NT Port Virtual Interface Device-Specific Error Checking and Debug 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	
		Device-Specific Error Status for Egress ECC Error
		Device-Specific Error Mask for Egress ECC Error
		ECC Error Check Disable
		Error Handler 32-Bit Error Status ( <i>Factory Test Only</i> )
		Error Handler 32-Bit Error Mask ( <i>Factory Test Only</i> )
		<i>Factory Test Only</i>
<i>Reserved</i>		Clock Enable
		Debug Control
		Power Management Hot Plug User Configuration (Legacy NT Mode)
		<i>Reserved</i> (NT PCI-to-PCI Bridge Mode)
		<i>Reserved</i>
		Bad TLP Counter
		Bad DLLP Counter
		<i>Reserved</i>
<i>Reserved</i>		Software Lane Status
		ACK Transmission Latency Limit (Legacy NT Mode)
		<i>Reserved</i> (NT PCI-to-PCI Bridge Mode)
		<i>Factory Test Only</i>

a. Certain registers are Port-specific, others are Chip-specific; all are Device-specific.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-36. 1DCh Debug Control**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> The first Configuration register programmed by the serial EEPROM must be the <b>Debug Control</b> register (Port 0, offset <b>IDCh</b> ), serial EEPROM locations 4h through 9h, as listed in <a href="#">Table 6-1</a> .				
If Port 0 is a Legacy NT Port, this register is loaded from the NT Port Virtual Interface register offset IDCh location in the serial EEPROM.				
In NT mode, all registers that are programmed by serial EEPROM for the Port that is the designated NT Port, must be programmed into the NT Port Virtual Interface (rather than the numbered Port).				
3:0	<b>Factory Test Only</b>	RO	No	Fh
4	<b>UPCFG Timer Enable</b> Reflects the STRAP_UPCFG_TIMER_EN# input state. 0 = Upconfigure Timer is disabled 1 = Upconfigure Timer is enabled	RWS	Yes	0 (STRAP_UPCFG_TIMER_EN#=H) 1 (STRAP_UPCFG_TIMER_EN#=L)
5	<b>SMBus Enable</b> Reflects the STRAP_SMBUS_EN# input state. 0 = SMBus Slave interface is disabled for device configuration (I <sup>2</sup> C mode is enabled) 1 = SMBus Slave interface is enabled for device configuration (SMBus mode is enabled)	RWS	Yes	0 (STRAP_SMBUS_EN#=H) 1 (STRAP_SMBUS_EN#=L)
6	<b>NT P2P Enable</b> Reflects the STRAP_NT_P2P_EN# input state. This bit and its corresponding input must <b>not</b> be toggled at runtime. 0 = NT PCI-to-PCI bridge mode is disabled (STRAP_NT_P2P_EN#=H) (Legacy NT mode is enabled) 1 = NT PCI-to-PCI bridge mode is enabled (STRAP_NT_P2P_EN#=L)	RWS	Yes	0 (STRAP_NT_P2P_EN#=H) 1 (STRAP_NT_P2P_EN#=L)
7	<b>Factory Test Only</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<p><b>Upstream Port ID</b>            Upstream Port Number – Reads the external Strap value on the <b>STRAP_UPSTRM_PORTSEL[3:0]</b> inputs, at Reset de-assertion.            When bit 15 (<i>Hardware/Software Configuration Mode Control</i>) is Cleared, software is not allowed to change this value.</p> <p>When bit 15 (<i>Hardware/Software Configuration Mode Control</i>) is Set, Upstream Port Number can be Set by software, using the values defined below.</p> <p><i>Note:</i> Port 0 is recommended for the upstream Port designation.</p> <p>0h = Port 0            1h = Port 1            2h = Port 2            3h = Port 3            4h = Port 4            5h = Port 5            6h = Port 6            7h = Port 7            8h = Port 8            9h = Port 9            Ah = Port 10            Bh = Port 11            Ch = Port 12            Dh = Port 13            Eh = Port 14            Fh = Port 15</p>	RO	Yes	
		RW <sup>a</sup>	Yes	Set by STRAP_UPSTRM_PORTSEL[3:0] input levels or by serial EEPROM, or by I <sup>2</sup> C followed by a Soft Reset

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
13:12	<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 8619 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 NT Port Virtual Interface, sends a Fatal Error Message to the Host, then asserts <b>FATAL_ERR#</b>.</li> <li>When the PEX 8619 detects an internal Fatal error (ECC failure), the switch sends a Fatal Interrupt Message to the Host and asserts <b>FATAL_ERR#</b>. 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)</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 NT Port Virtual Interface logs the error in the NT Port Virtual Interface Error Status register, then asserts <b>FATAL_ERR#</b>. This Fatal error detection blocks all 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 all the Mode 3 actions, the PEX 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14	<b>Factory Test Only</b>	RWS	Yes	0
15	<p><b>Hardware/Software Configuration Mode Control</b> Allows software to configure which Port is the upstream Port, as well as which Port is a Legacy NT Port.</p> <p>0 = Upstream Port and NT Port selection by the <b>STRAP_UPSTRM_PORTSEL[3:0]</b> and <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs, respectively, which can be overridden by the serial EEPROM and/or I<sup>2</sup>C configuration mechanism. Cannot be changed by in-band software during runtime.</p> <p>1 = In-band software can change which Port is configured to be the upstream Port and NT Port, by writing new values to fields [11:8 and 27:24] (<b>Upstream Port ID</b> and <b>NT Port Number</b>, respectively), followed by issuance of a Hot Reset to the upstream Port. Bit 20 (<b>Upstream Port and NT-Link Port DL_Down Reset Propagation Disable</b>) must be Cleared.</p>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<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 r2.0</i> <i>Note:</i> Only a Fundamental Reset serial EEPROM load affects this bit.	RWS	Yes	1
17	<b>Disable Serial EEPROM Load on Hot Reset</b> 0 = Enables serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state 1 = Disables serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state	RWS	Yes	0
18	<b>NT Mode Enable</b> NT mode (Intelligent Adapter) is enabled by the <b>STRAP_NT_ENABLE#</b> input, which is overridden by the value of this bit if this register is programmed by the serial EEPROM upon Fundamental Reset. Software, serial EEPROM load upon upstream Port or NT Port Link Interface Hot Reset or <i>DL_Down</i> state, and/or I <sup>2</sup> C are not allowed to change this value. 0 = NT mode is disabled (STRAP_NT_ENABLE#=H) 1 = NT mode is enabled (STRAP_NT_ENABLE#=L)	HwInit	Yes (Serial EEPROM only)	Set by STRAP_NT_ENABLE# input level or by serial EEPROM
19	<b>NT Port DL_Down Reset Propagation Disable</b> Setting this bit: <ul style="list-style-type: none"> <li>Enables the upstream Port to ignore a Hot Reset training sequence,</li> <li>Blocks the PEX 8619 from manifesting an internal reset due to a <i>DL_Down</i> event, 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<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 Port and NT-Link Port to ignore a Hot Reset training sequence,</li> <li>• Blocks the PEX 8619 from manifesting an internal reset due to a DL_Down event, and</li> <li>• Blocks the PEX 8619 NT Port Link Interface from manifesting an internal reset due to a DL_Down 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 DL_Down event occurs on the upstream Link</li> </ul> <b>This bit typically should be Set for NT mode.</b>	RWS	Yes	0
21	<b>Cut-Thru Enable</b> 0 = Disables Cut-Thru support 1 = Enables Cut-Thru support	RWS	Yes	1
22	<b>Reserved</b>	RsvdP	No	0
23	<b>Factory Test Only</b>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
	<p><b>NT Port Number</b></p> <p>When bit 18 (<i>NT Mode Enable</i>) is Set, and bit 15 (<i>Hardware/Software Configuration Mode Control</i>) is Cleared, the NT Port Number is Set by the <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs.</p> <p>This field is “Don’t Care” for Transparent mode. Software is not allowed to change this value.</p>	HwInit	Yes	
27:24	<p>When bits 18 (<i>NT Mode Enable</i>) and 15 (<i>Hardware/Software Configuration Mode Control</i>) are both Set, the NT Port Number selected by this field is Set by software, using the values defined below.</p> <p>0h = Port 0      1h = Port 1      2h = Port 2      3h = Port 3      4h = Port 4      5h = Port 5      6h = Port 6      7h = Port 7      8h = Port 8      9h = Port 9      Ah = Port 10      Bh = Port 11      Ch = Port 12      Dh = Port 13      Eh = Port 14      Fh = Port 15</p>	R/W <sup>a</sup>	Yes	Set by STRAP_NT_UPSTRM_PORTSEL[3:0] input levels or by serial EEPROM, or by I <sup>2</sup> C followed by a Soft Reset

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<p><b>Virtual Interface Access Enable</b>  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</p> <p><i>Notes:</i> This bit does not affect the PEX 8619 in Transparent mode, nor does it affect other transaction types.</p> <p><i>Set this bit to enable Configuration access to the NT Port Virtual Interface.</i></p>	RW	Yes	1
29	<p><b>Link Interface Access Enable</b>  0 = Retries Type 0 Configuration Request received on the NT Port Link Interface  1 = Accepts Type 0 Configuration Request received on the NT Port Link Interface</p> <p><i>Notes:</i> This bit does not affect the PEX 8619 in Transparent mode.</p> <p><i>Set this bit to enable Configuration access to the NT Port Link Interface.</i></p>	RW	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-36. 1DCh Debug Control****(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
30	<b>Inhibit EEPROM NT-Link Load on Hot Reset</b> Inhibits serial EEPROM load of NT Port Link Interface registers when any one of the following conditions exist: <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [17:16] (<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 [20, 17:16] (<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 17 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul>	RW	Yes	1
31	<b>Load Only EEPROM NT-Link on Hot Reset</b> Load only serial EEPROM NT Port Link Interface register entries when any one of the following conditions exist: <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [17:16] (<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 [20, 17:16] (<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 17 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul>	RW	Yes	1

a. Although these bits are RW, do not change by software.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-37. 1E0h Power Management Hot Plug User Configuration**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is reserved in NT PCI-to-PCI Bridge mode.				
11:0	<b>Reserved</b>	RsvdP	No	0-0h
12	<b>NT Hot Plug Enable</b> Can be programmed only by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	0
16:13	<b>Reserved</b>	RsvdP	No	0-0h
17	<b>40-Pin I/O Expander Enable</b> 0 = Enables 16-pin I/O Expanders for all downstream Ports that implement Serial Hot Plug 1 = Enables 40-pin I/O Expanders for all downstream Ports that implement Serial Hot Plug <i>Note:</i> Value of 1 can be enabled only by serial EEPROM (that is, neither software nor I <sup>2</sup> C can be used to enable the 40-Pin I/O Expander).	RW	Yes (See Note)	0
31:18	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-38. 1E8h Bad TLP Counter**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad TLP Counter</b> Value reflects errors detected on the NT Port Link Interface. Counts the quantity of TLPs received with bad Link Cyclic Redundancy Check (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 16-39. 1ECh Bad DLLP Counter**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad DLLP Counter</b> Value reflects errors detected on the NT Port Link Interface. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-40. 1F8h ACK Transmission Latency Limit**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
The value of this register should be valid after Link negotiation.				
<i>Note:</i> This register is reserved in NT PCI-to-PCI Bridge mode.				
11:0	<b>ACK Transmission Latency Limit</b> Acknowledge Control Packet (ACK) Transmission Latency Limit. The value of this field changes, based upon <a href="#">Negotiated Link Width</a> (offset 78h[25:20]) encoding after the Link is up. x1 Link width = 255d x2 Link width = 217d x4 Link width = 118d x8 Link width = 107d	RWS	Yes	0EDh
15:12	<b>Reserved</b>	RsvdP	No	0h
23:16	<b>Upper 8 Bits of the Replay Timer Limit</b> The value in this register is a multiplier of the default internal timer values that are compliant to the <i>PCI Express Base r2.0</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 16.13.2 NT Port Virtual Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)

The registers detailed in [Section 14.16.5, “Device-Specific Registers – Ingress Control and Port Enable \(Offsets 660h – 67Ch\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-13](#) (register map), and [Register 16-41](#) and [Register 16-42](#).

**Table 16-13. NT Port Virtual Interface Device-Specific Ingress Control and Port Enable 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	
Ingress Control				660h
Ingress Control Shadow				664h
<i>Reserved</i>		Port Enable Status		
Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7				668h
Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15				66Ch
<i>Reserved</i>		Port Cut-Thru Enable Status		
Ingress PLL RAM ECC 1-Bit Counter				670h
Ingress Control 2 <i>Factory Test Only</i>				674h
Ingress Control 2 <i>Factory Test Only</i>				678h
Ingress Control 2 <i>Factory Test Only</i>				67Ch

### Register 16-41. 660h Ingress Control

(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:0	<i>Factory Test Only</i>	RWS	Yes	0-0h
26	<b>Disable Upstream Port BAR0 and BAR1</b> 0 = Enables the upstream Port <b>Base Address 0</b> and <b>Base Address 1</b> registers ( <b>BAR0</b> and <b>BAR1</b> , offsets 10h and 14h, respectively) 1 = Disables the upstream Port <b>Base Address 0</b> and <b>Base Address 1</b> registers ( <b>BAR0</b> and <b>BAR1</b> , offsets 10h and 14h, respectively)	RWS	Yes	0
27	<i>Not used</i>	RWS	Yes	0
28	<b>Disable VGA BIOS Memory Access Decoding</b> 0 = Enables the <b>Bridge Control</b> register <b>VGA 16-Bit Decode Enable</b> , <b>VGA Enable</b> , and <b>ISA Enable</b> bits (offset 3Ch[20:18], respectively), and enables decoding of the PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are blocked) 1 = Disables the <b>Bridge Control</b> register <b>VGA 16-Bit Decode Enable</b> , <b>VGA Enable</b> , and <b>ISA Enable</b> bits (offset 3Ch[20:18], respectively), and disables decoding of the PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are <i>not</i> blocked)	RWS	Yes	0
30:29	<i>Factory Test Only</i>	RWS	Yes	00b
31	<i>Not used</i>	RWS	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-42. 664h Ingress Control Shadow  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Factory Test Only</b>	RWS	Yes	0
1	<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
3:2	<b>Factory Test Only</b>	RWS	Yes	00b
4	<b>Drop EP TLPs</b> Drop Endpoint TLPs.  1 = Endpoint TLP was dropped	RWS	Yes	0
5	<b>No Special Treatment for Relaxed Ordering Traffic</b> The PEX 8619 supports Relaxed Ordering for Completions. By default, if the RO attribute is Set within a Completion, then 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.  1 = Device-Specific Relaxed Ordering Completion will not be flagged to egress block	RWS	Yes	0
7:6	<b>Factory Test Only</b>	RWS	Yes	00b
8	<b>Drop ECRC TLPs</b> Drop End-to-end Cyclic Redundancy Check (ECRC) TLPs.  1 = ECRC TLP was dropped	RWS	Yes	0
10:9	<b>ACK TLP Counter Timeout</b> Sets the quantity of ingress TLP Acknowledgements (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
31:11	<b>Factory Test Only</b>	RWS	Yes	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 16.13.3 NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)

The registers detailed in [Section 14.16.10, “Device-Specific Registers – Vendor-Specific Extended Capability 2 \(Offsets 950h – 95Ch\),”](#) are also applicable to the NT Port, except for the Next Capability Offset value, as defined in [Table 16-14](#) (register map) and [Register 16-43](#).

**Table 16-14. NT Port Virtual Interface Device-Specific, Vendor-Specific Extended Capability 2 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 2 (1h)	PCI Express Extended Capability ID 2 (000Bh)	950h
Vendor-Specific Header 2			954h
Hardwired Device ID		Hardwired Vendor ID	958h
<i>Reserved</i>		PLX Hardwired Revision ID	95Ch

**Register 16-43. 950h Vendor-Specific Extended Capability 2**

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, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	RO	Yes	000Bh
19:16	<b>Capability Version 2</b>	RO	Yes	1h
31:20	<b>Next Capability Offset 2</b> Program to C34h, which addresses the <b>Vendor-Specific Extended Capability 4</b> structure.	RO	Yes	C34h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 16.13.4 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 16-15 defines the register map used by the NT Port Virtual Interface.

**Table 16-15. 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 (000h)	Capability Version 4 (1h)	PCI Express Extended Capability ID 4 (000Bh)
	Vendor-Specific Header 4	C34h
	Memory BAR2 Address Translation Lower	C38h
	Memory BAR3 Address Translation Upper	C3Ch
	Memory BAR4 Address Translation Lower	C40h
	Memory BAR5 Address Translation Upper	C44h
<i>Reserved</i>	Virtual Interface IRQ Set	C48h
<i>Reserved</i>	Virtual Interface IRQ Clear	C4Ch
<i>Reserved</i>	Virtual Interface IRQ Mask Set	C50h
<i>Reserved</i>	Virtual Interface IRQ Mask Clear	C54h
<i>Reserved</i>	Link Interface IRQ Set	C58h
<i>Reserved</i>	Link Interface IRQ Clear	C5Ch
<i>Reserved</i>	Link Interface IRQ Mask Set	C60h
<i>Reserved</i>	Link Interface IRQ Mask Clear	C64h
	NT Port SCRATCH0	C68h
	NT Port SCRATCH1	C6Ch
	NT Port SCRATCH2	C70h
	NT Port SCRATCH3	C74h
	NT Port SCRATCH4	C78h
	NT Port SCRATCH5	C7Ch
	NT Port SCRATCH6	C80h
	NT Port SCRATCH7	C84h
		C88h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-44. C34h Vendor-Specific Extended Capability 4**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
15:0	<b>PCI Express Extended Capability ID 4</b> Program to 000Bh, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	RO	Yes	000Bh
19:16	<b>Capability Version 4</b>	RO	Yes	1h
31:20	<b>Next Capability Offset 4</b> 000h = This extended capability is the last capability in the PEX 8619 Extended Capabilities list	RO	Yes	000h

**Register 16-45. C38h Vendor-Specific Header 4**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
15:0	<b>Vendor-Specific ID 4</b> ID Number of this Extended Capability structure.	RO	Yes	0003h
19:16	<b>Vendor-Specific Rev 4</b> Version Number of this structure.	RO	Yes	0h
31:20	<b>Vendor-Specific Length 4</b> Quantity of bytes in the entire structure.	RO	Yes	078h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 16-46. C3Ch Memory BAR2 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	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, offset D4h[31], is Set).	RW	Yes	000h

**Register 16-47. C40h Memory BAR3 Address Translation Upper**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	Offset D8h[31]=0	RsvdZ	No
	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
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 <b>BAR3 Enable</b> bit, offset D8h[31], is Set).	RW	Yes	000h

**Register 16-48. C44h Memory BAR4 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	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, offset DCh[31], is Set).	RW	Yes	000h

**Register 16-49. C48h Memory BAR5 Address Translation Upper**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<b>Reserved</b>	Offset E0h[31]=0	RsvdZ	No
	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
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 <b>BAR5 Enable</b> bit, offset E0h[31], is Set).	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-50. C4Ch Virtual Interface IRQ Set**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> The bits in this register can be masked by their respective <b>Virtual Interface IRQ Mask Set</b> 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 if the following conditions exist: <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 16-51. C50h Virtual Interface IRQ Clear**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> The bits in this register can be masked by their respective <b>Virtual Interface IRQ Mask Clear</b> 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 if the following conditions exist: <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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-52. 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	<p><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</p>	RW1S	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 16-53. 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	<p><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</p>	RW1C	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-54. C5Ch Link Interface IRQ Set**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> The bits in this register can be masked by their respective <a href="#">Link Interface IRQ Mask Set</a> 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 if the following conditions exist: <ul style="list-style-type: none"> <li>• This register (offset C5Ch or C60h) value is non-zero, and,</li> <li>• Corresponding <a href="#">Link Interface IRQ Mask Set</a> or <a href="#">Link Interface IRQ Mask Clear</a> 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 16-55. C60h Link Interface IRQ Clear**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> The bits in this register can be masked by their respective <a href="#">Link Interface IRQ Mask Clear</a> 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 if the following conditions exist: <ul style="list-style-type: none"> <li>• This register (offset C60h or C5Ch) value is zero (0), –or–</li> <li>• <a href="#">Link Interface IRQ Mask Set</a> or <a href="#">Link Interface IRQ Mask Clear</a> 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-56. 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 16-57. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-58. 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.	RW	Yes	0000_0000h

**Register 16-59. 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.	RW	Yes	0000_0000h

**Register 16-60. 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.	RW	Yes	0000_0000h

**Register 16-61. 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.	RW	Yes	0000_0000h

**Register 16-62. 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.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-63. 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.	RW	Yes	0000_0000h

**Register 16-64. 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.	RW	Yes	0000_0000h

**Register 16-65. 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.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 16.14 NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – EECh)

Table 16-16 defines the register map of the NT Port Virtual Interface NT Bridging-Specific registers.

**Table 16-16. 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	
	<i>Reserved</i>	C8Ch – D64h
NT Port Virtual Interface NT Bridging-Specific Registers – Base Address and Base Address Setup (Offsets D68h – D90h) (Shadow Copy) (Legacy NT Mode)		D68h
	<i>Reserved</i> (NT PCI-to-PCI Bridge Mode)	...
		D90h
NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DB0h)		D94h
		...
		DB0h
	<i>Reserved</i>	DB4h – EECh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 16.14.1 NT Port Virtual Interface NT Bridging-Specific Registers – Base Address and Base Address Setup (Offsets D68h – D90h)

The registers in this section are shadow copies of the NT Port Virtual Interface Base Address register (BAR) and BAR Configuration registers, and valid only for Port 0. If Port 0 is the NT Port, the registers are in Virtual Interface Configuration Space. [Table 16-17](#) defines the register map.

**Notes:** *This register structure is reserved in NT PCI-to-PCI Bridge mode.*

*It is recommended that these registers not be changed from their default values.*

**Table 16-17. NT Port Virtual Interface NT Bridging-Specific Base Address Register (BAR) and BARx Setup Register Map (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
		BAR0 (Shadow Copy)
		BAR1 (Shadow Copy)
		BAR2 (Shadow Copy)
		BAR3 (Shadow Copy)
		BAR4 (Shadow Copy)
		BAR5 (Shadow Copy)
		BAR0/1 Setup (Shadow Copy)
		Memory BAR2 Setup (Shadow Copy)
		Memory BAR2/3 Setup (Shadow Copy)
		Memory BAR4 Setup (Shadow Copy)
		Memory BAR4/5 Setup (Shadow Copy)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-66. D68h BAR0 (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 0 (BAR0)</b> register (offset 10h).				
0	<b>Memory Space Indicator</b> Virtual Side <b>BAR0</b> is configured by serial EEPROM and the Local Host. By default, Configuration <b>BAR0/1 Setup</b> register selects 32-bit Memory <b>BAR0</b> . 0 = Memory space only supported	RsvdP	No	0
2:1	<b>Memory Map Type</b> Reflection of the BAR configuration, as specified in the <b>BAR0/1 Setup (Shadow Copy)</b> register (offset D80h). 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	0-0h
31:17	<b>Base Address 0</b> 128-KB Base address in which to map the PEX 8619 Configuration Space registers into Memory space.	RW	Yes	0-0h

**Register 16-67. D6Ch BAR1 (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 1 (BAR1)</b> register (offset 14h).				
31:0	<b>Upper 32-Bit BAR0 of Virtual NT Port</b> <b>BAR1</b> is the upper 32 bits of 64-bit <b>BAR0/1</b> , Base Address 1 extends Base Address 0 to provide the upper 32 Address bits when the <b>BAR0 (Shadow Copy)</b> register <i>Memory Map Type</i> field (offset D68h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	<b>Reserved</b> when the <b>BAR0 (Shadow Copy)</b> register is not enabled as a 64-bit BAR ( <i>Memory Map Type</i> field (offset D68h[2:1]) is not programmed to 10b).	RsvdP	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-68. D70h BAR2 (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 2 (BAR2)</b> register (offset 18h).				
0	<b>Memory Space Indicator</b> 0 = Memory space only supported	RO	No	0
2:1	<b>Memory Map Type</b> Reflection of the BAR configuration, as specified in the <b>Memory BAR2 Setup (Shadow Copy)</b> register (offset D84h).  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	0_000h
31:20	<b>Base Address 2</b>	RW	Yes	000h

**Register 16-69. D74h BAR3 (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 3 (BAR3)</b> register (offset 1Ch).				
This register has RW privilege if <b>BAR2/3</b> is configured as a 64-bit BAR ( <b>BAR2 (Shadow Copy)</b> register <b>Memory Map Type</b> field, offset <b>D70h[2:1]</b> , is programmed to 10b).				
0	<b>Memory Space Indicator</b> 0 = Memory space only supported	Offset D70h[2:1]=00b Offset D70h[2:1]=10b	RsvdP RW	No 0
2:1	<b>Memory Map Type</b> Reflection of the BAR configuration, as specified in the <b>Memory BAR2/3 Setup (Shadow Copy)</b> register (offset D88h).  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 D70h[2:1]=00b Offset D70h[2:1]=10b	RsvdP RW	No 00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset D70h[2:1]=00b Offset D70h[2:1]=10b	RsvdP RW	No 0
19:4	<b>Reserved</b> 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 D70h[2:1]=00b Offset D70h[2:1]=10b	RsvdP RW	No 0_000h
31:20	<b>Base Address 3</b>	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-70. D78h BAR4 (Shadow Copy)  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 4 (BAR4)</b> register (offset 20h).				
0	<b>Memory Space Indicator</b> 0 = Memory space only supported	RO	No	0
2:1	<b>Memory Map Type</b> Reflection of the BAR configuration, as specified in the <b>Memory BAR4 Setup (Shadow Copy)</b> register (offset D8Ch).  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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	0_000h
31:20	<b>Base Address 4</b>	RW	Yes	000h

**Register 16-71. D7Ch BAR5 (Shadow Copy)  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i>Notes:</i> This register is a shadow copy of the NT Port Virtual Interface <b>Base Address 5 (BAR5)</b> register (offset 24h).  This register has RW privilege if <b>BAR4/5</b> is configured as a 64-bit BAR ( <b>BAR4 (Shadow Copy)</b> register <b>Memory Map Type</b> field, offset <b>D78h[2:1]</b> , is programmed to 10b).					
0	<b>Memory Space Indicator</b> 0 = Memory space only supported	Offset D78h[2:1]=00b Offset D78h[2:1]=10b	RsvdP RW	No Yes	0 0
2:1	<b>Memory Map Type</b> Reflection of the BAR configuration, as specified in the <b>Memory BAR4/5 Setup (Shadow Copy)</b> register (offset D90h).  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 D78h[2:1]=00b Offset D78h[2:1]=10b	RsvdP RW	No Yes	00b 00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset D78h[2:1]=00b Offset D78h[2:1]=10b	RsvdP RW	No Yes	0 0
19:4	<b>Reserved</b>  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 D78h[2:1]=00b Offset D78h[2:1]=10b	RsvdP RW	No Yes	0_000h 0_000h
31:20	<b>Base Address 5</b>	RW	Yes	000h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-72. D80h BAR0/1 Setup (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register must be programmed with the same value as the <b>NT Port Virtual Interface BAR0/1 Setup</b> register (offset <b>D0h</b> ).				
1:0	<b>Type Selector</b> 00b = Disables Virtual Interface <b>BAR0</b> and <b>BAR1</b> 01b = <b>Reserved</b> 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)	RW	Yes	10b
2	<b>BAR0 Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
31:3	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-73. D84h Memory BAR2 Setup (Shadow Copy)**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register must be programmed with the same value as the <b>NT Port Virtual Interface Memory BAR2 Setup</b> register (offset <b>D4h</b> ). This requirement applies only to the NT Port Virtual Interface.				
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.	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	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 0, and Writes are ignored 1 = Corresponding <b>BAR2</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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] ( <b>BAR2 Type</b> ) = 00b	RW	Yes
	<b>BAR2 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <b>BAR2 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-74. D88h Memory BAR2/3 Setup (Shadow Copy)  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register must be programmed with the same value as the <b>NT Port Virtual Interface Memory BAR2/3 Setup</b> register (offset <b>D8h</b> ). This requirement applies only to the NT Port Virtual Interface.				
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.	RO	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	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 0, and Writes are ignored 1 = Corresponding <b>BAR3</b> bits are RW bits  <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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 D84h[2:1] ( <b>BAR2 Type</b> ) = 00b	RW	Yes
	<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			

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-75. D8Ch Memory BAR4 Setup (Shadow Copy)  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register must be programmed with the same value as the <b>NT Port Virtual Interface Memory BAR4 Setup</b> register (offset <b>DCh</b> ). This requirement applies only to the NT Port Virtual Interface.				
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 10b = <b>BAR4/5</b> is implemented as a 64-bit Memory BAR	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	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 0, and Writes are ignored 1 = Corresponding <b>BAR4</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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] ( <b>BAR4 Type</b> ) = 00b	RW	Yes
	<b>BAR4 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <b>BAR4 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-76. D90h Memory BAR4/5 Setup (Shadow Copy)  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register must be programmed with the same value as the <b>NT Port Virtual Interface Memory BAR4/5 Setup</b> register (offset E0h). This requirement applies only to the NT Port Virtual Interface.				
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.	RO	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	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 0, and Writes are ignored 1 = Corresponding <b>BAR5</b> bits are RW bits <i>Note:</i> If <b>BAR[30:n]</b> is the Base field (BAR size is $2^n$ ), <b>BAR[30:n]</b> should have all ones (1).	RW	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 D8Ch[2:1] ( <b>BAR4 Type</b> ) = 00b	RW	Yes
	<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 D8Ch[2:1] ( <b>BAR4 Type</b> ) = 10b	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 16.14.2 NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DB0h)

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* bit (bit 31) of each register accordingly.

[Table 16-18](#) defines the register and address locations, as they relate to [Register 16-77](#).

**Table 16-18. NT Port Virtual Interface NT Bridging-Specific Requester ID Translation Lookup Table Entry\_n Register Locations**

ADDR Location	Lookup Table Entry_n	ADDR Location	Lookup Table Entry_n
D94h	0	DA4h	4
D98h	1	DA8h	5
D9Ch	2	DACh	6
DA0h	3	DB0h	7

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-77. D94h – DB0h NT Port Virtual Interface Requester ID Translation  
LUT Entry<sub>\_n</sub> (where n = 0 through 7)**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default	
2:0	<b>Requester ID on Virtual Side</b>		<b>Function Number</b> LUT Entry <sub>_n</sub> Requester Function Number.	RW	Yes	000b
7:3			<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
29:16	<b>Reserved</b>		RsvdP	No	0-0h	
30	<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	
31	<b>LUT Entry<sub>_n</sub> Enable</b>  0 = Disables 1 = Enables		RW	Yes	0	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.15 NT Port Virtual Interface Device-Specific Registers (Offsets E40h – F8Ch)

The registers detailed in [Section 14.16, “Device-Specific Registers \(Offsets 530h – F8Ch\)”](#) (for offsets E40h through F8Ch), are unique to the PEX 8619 and not referenced in the *PCI Express Base r2.0*. These registers are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-19](#) (register map) and [Table 16-22](#), and [Register 16-78](#) through [Register 16-93](#).

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- [Section 16.13, “NT Port Virtual Interface Device-Specific Registers \(Offsets 1C0h – C88h\)”](#)
- [Section 16.17, “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 16-19. NT Port Virtual Interface Device-Specific Register Map (Offsets E40h – F8Ch)**

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	
		E40h
		...
		EFCh
		F00h
		...
		F3Ch
		F40h
		...
		F4Ch
		F50h
		...
		F8Ch
NT Port Virtual Interface Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)		
NT Port Virtual Interface Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch)		
Device-Specific Registers – Error Reporting (Offsets F40h – F4Ch)		
NT Port Virtual Interface Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch)		

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 16.15.1 NT Port Virtual Interface Device-Specific Registers – Physical Layer (Offsets E40h – EFCh)

A Physical Layer (PHY) register detailed in [Section 14.16.16, “Device-Specific Registers – Physical Layer \(Offsets E40h – EFCh\),”](#) is also applicable to the NT Port Virtual Interface, except as defined in [Table 16-20](#) (register map; offsets E40h through E48h are *reserved*).

**Table 16-20. NT Port Virtual Interface Device-Specific PHY 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>	E40h – EECh
	Physical Layer Control	EF0h
	<i>Reserved</i>	EF4h – EFCh

### 16.15.2 NT Port Virtual Interface Device-Specific Registers – Source Queue Weight and Soft Error (Offsets F00h – F3Ch)

Device-Specific registers detailed in [Section 14.16.17, “Device-Specific Registers – Source Queue Weight and Soft Error \(Offsets F00h – F3Ch\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-21](#) (register map; offsets F10h, F18h, and F24h through F30h are *reserved*).

**Table 16-21. NT Port Virtual Interface Source Queue Weight and Soft Error 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	
	ECRC Check Enable for Ports 0, 1, 2, 3	F00h
	ECRC Check Enable for Ports 4, 5, 6, 7	F04h
	ECRC Check Enable for Ports 8, 9, 10, 11	F08h
	ECRC Check Enable for Ports 12, 13, 14, 15	F0Ch
	<i>Reserved</i>	F10h – F3Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 16.15.3 NT Port Virtual Interface Device-Specific Registers – ARI Capability (Offsets F50h – F8Ch)

The Alternative Routing-ID Interpretation (ARI) Capability registers detailed in [Section 14.16.19, “Device-Specific Registers – ARI Capability \(Offsets F50h – F8Ch\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 16-22](#) (register map), and [Register 16-78](#) through [Register 16-93](#).

**Table 16-22. NT Port Virtual Interface Device-Specific ARI Capability Register Map  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT 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	
Device Control 2 for Port 0	Device Capability 2 for Port 0	F50h
Device Control 2 for Port 1	Device Capability 2 for Port 1	F54h
Device Control 2 for Port 2	Device Capability 2 for Port 2	F58h
Device Control 2 for Port 3	Device Capability 2 for Port 3	F5Ch
Device Control 2 for Port 4	Device Capability 2 for Port 4	F60h
Device Control 2 for Port 5	Device Capability 2 for Port 5	F64h
Device Control 2 for Port 6	Device Capability 2 for Port 6	F68h
Device Control 2 for Port 7	Device Capability 2 for Port 7	F6Ch
Device Control 2 for Port 8	Device Capability 2 for Port 8	F70h
Device Control 2 for Port 9	Device Capability 2 for Port 9	F74h
Device Control 2 for Port 10	Device Capability 2 for Port 10	F78h
Device Control 2 for Port 11	Device Capability 2 for Port 11	F7Ch
Device Control 2 for Port 12	Device Capability 2 for Port 12	F80h
Device Control 2 for Port 13	Device Capability 2 for Port 13	F84h
Device Control 2 for Port 14	Device Capability 2 for Port 14	F88h
Device Control 2 for Port 15	Device Capability 2 for Port 15	F8Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-78. F50h Device Capability and Control 2 for Port 0**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 0</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes
		Otherwise	RO	Yes
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 0</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-79. F54h Device Capability and Control 2 for Port 1**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 1</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes
		Otherwise	RO	Yes
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 1</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-80. F58h Device Capability and Control 2 for Port 2  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 2</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 2</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-81. F5Ch Device Capability and Control 2 for Port 3  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 3</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 3</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-82. F60h Device Capability and Control 2 for Port 4  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 4</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 4</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-83. F64h Device Capability and Control 2 for Port 5  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 5</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 5</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-84. F68h Device Capability and Control 2 for Port 6  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 6</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 6</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-85. F6Ch Device Capability and Control 2 for Port 7  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 7</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 7</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 16-86. F70h Device Capability and Control 2 for Port 8**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 8</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 8</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-87. F74h Device Capability and Control 2 for Port 9**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 9</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 9</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-88. F78h Device Capability and Control 2 for Port 10  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 10</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes 0	
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 10</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-89. F7Ch Device Capability and Control 2 for Port 11  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 11</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0. 0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes 0	
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 11</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-90. F80h Device Capability and Control 2 for Port 12**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 12</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0.  0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <b>ARI Forwarding Supported</b> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 12</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-91. F84h Device Capability and Control 2 for Port 13**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 13</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0.  0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <b>ARI Forwarding Supported</b> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
		Otherwise	RO	Yes 0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 13</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 16-92. F88h Device Capability and Control 2 for Port 14**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 14</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0.  0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 14</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 16-93. F8Ch Device Capability and Control 2 for Port 15**  
**(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Capability 2 for Port 15</b>				
4:0	<b>Reserved</b>	RsvdP	No	0-0h
5	<b>ARI Forwarding Supported</b> If the NT Port is Port 0, the NT Port Virtual Interface holds the ARI forwarding supported, to a value of 1. Otherwise, the default value is 0.  0 = ARI forwarding is not supported 1 = ARI forwarding is supported when the <b>Device Capability 2</b> register <i>ARI Forwarding Supported</i> bit (offset 8Ch[5]) is Set for this Port	NT Port is Port 0	RO	Yes 1
	Otherwise	RO	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Control 2 for Port 15</b>				
20:16	<b>Reserved</b>	RsvdP	No	0-0h
21	<b>ARI Forwarding Enable</b> 1 = Configuration Type 1-to-Type 0 occurs, even if the Device Number is not 0	RW	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 16.16 NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

The registers detailed in [Section 14.17, “Advanced Error Reporting Extended Capability Registers \(Offsets FB4h – FDCh\),”](#) are also applicable to the NT Port Virtual Interface, except for the [Next Capability Offset](#) value, as defined in [Table 16-23](#) (register map), [Register 16-94](#), and the exceptions indicated in the note.

**Notes:** *Data Link Protocol and Surprise Down PCI Express errors are not valid for the NT Port Virtual nor Link Interfaces, and therefore do not affect register offsets FB8h, FBCh, and FC0h in NT mode.*

*Receiver, Bad TLP, Bad DLLP, Replay Number Rollover, and Replay Timer Timeout PCI Express errors are not valid for the NT Port Virtual nor Link Interfaces, and therefore do not affect register offsets FC4h and FC8h in NT mode.*

**Table 16-23. 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
		Uncorrectable Error Status	FB8h
		Uncorrectable Error Mask	FBCh
		Uncorrectable Error Severity	FC0h
		Correctable Error Status	FC4h
		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 16-94. 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	RO	Yes	0001h
19:16	Capability Version	RO	Yes	1h
31:20	Next Capability Offset Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	RO	Yes	148h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 16.17 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 r2.0*. [Table 16-24](#) defines the register map used by the NT Port Virtual Interface.

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- [Section 16.13, “NT Port Virtual Interface Device-Specific Registers \(Offsets 1C0h – C88h\)”](#)
- [Section 16.15, “NT Port Virtual Interface Device-Specific Registers \(Offsets E40h – F8Ch\)”](#)

*Note: It is recommended that these registers not be changed from their default values.*

**Table 16-24. NT Port Virtual Interface Device-Specific Register Map – Link Error (Offsets FE0h – FFCh) (Port 0, when Port 0 is the NT 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	
	Link Error Status Virtual	FE0h
	Link Error Mask Virtual	FE4h
	<i>Reserved</i>	FE8h – FFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 16-95. FE0h Link Error Status Virtual  
(Port 0, when Port 0 is the NT Port)**

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 <a href="#">Link Error Mask Virtual</a> register bits (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE4h</b> ).				
0	<b>Link Side Correctable Error Status</b> 1 = NT Port Link Interface detected a Correctable TLP error, and signaled the interrupt to the Local 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 Local 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 Local Host	RW1CS	Yes	0
31:4	<b>Reserved</b>	RsvdP	No	0000_000h

**Register 16-96. FE4h Link Error Mask Virtual  
(Port 0, when Port 0 is the NT Port)**

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 <a href="#">Link Error Status Virtual</a> register bits (Port 0, when Port 0 is the NT Port, Virtual Interface Only, offset <b>FE0h</b> ).				
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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 17 NT Port Link Interface Registers – NT Mode Only

### 17.1 Introduction

*Note:* Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

The NT Port includes two sets of Configuration, Capability, Control, and Status registers, to support the Link and Virtual Interfaces. This chapter defines the NT Port Link Interface registers. Other registers are defined in:

- Chapter 14, “Transparent Port Registers”
- Chapter 16, “NT Port Virtual Interface Registers – NT Mode Only”
- Chapter 18, “DMA Interface Registers – DMA Mode Only”

*Note:* For Chip-specific registers (those that exist only in Port 0), if Port 0 is a Legacy NT Port, then those registers exist only in the NT Port Virtual Interface.

All PEX 8619 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 r2.0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

953

## 17.2 NT Port Link Interface Type 0 Register Map

Table 17-1 defines the NT Port Link Interface Type 0 register mapping.

**Table 17-1. 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	
		00h
		...
		34h
		...
NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)	Capability Pointer (40h)	3Ch
		40h
		44h
	Next Capability Pointer (48h)      Capability ID (01h)	48h
NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h)		...
	Next Capability Pointer (68h)      Capability ID (05h)	64h
NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h)		68h
	Next Capability Pointer (A4h)      Capability ID (10h)	...
NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)		A0h
	Next Capability Pointer (C8h)      SSID/SSVID Capability ID (0Dh)	A4h
NT Port Link Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)		...
	Next Capability Pointer 3 (00h)      PCI Express Capability ID 3 (09h)	C4h
NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)		C8h
Next Capability Offset (FB4h)	1h	...
	PCI Express Extended Capability ID (0003h)	FCh
NT Port Link Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)		100h
Next Capability Offset (148h)	1h	...
	PCI Express Extended Capability ID (0004h)	134h
NT Port Link Interface Power Budget Extended Capability Registers (Offsets 138h – 144h)		138h
Next Capability Offset (950h)	1h	...
	PCI Express Extended Capability ID (0002h)	144h
NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)		148h
		...
		1BCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 17-1. NT Port Link Interface Type 0 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	
		NT Port Link Interface Device-Specific Registers (Offsets 1C0h – C88h)
Next Capability Offset 2 (C34h)	1h	PCI Express Extended Capability ID 2 (000Bh)
		NT Port Link Interface Device-Specific Registers (Offsets 1C0h – C88h)
Next Capability Offset 4 (000h)	1h	PCI Express Extended Capability ID 4 (000Bh)
		NT Port Link Interface Device-Specific Registers (Offsets 1C0h – C88h)
		NT Port Link Interface NT Bridging-Specific Registers (Offsets C8Ch – EFCh)
		NT Port Link Interface Device-Specific Registers – Source Queue Weight (Offsets F00h – F3Ch)
		<i>Reserved</i>
Next Capability Offset (138h)	1h	PCI Express Extended Capability ID (0001h)
		NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)
		<i>Reserved</i>
		FE0h – FFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.3 Register Access

The PEX 8619 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 8619 supports three mechanisms for accessing the NT Port Link Interface registers:

- [PCI Express Base r2.0 Configuration Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- [Device-Specific Cursor Mechanism](#)

### 17.3.1 [PCI Express Base r2.0 Configuration Mechanism](#)

The *PCI Express Base r2.0* 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 8619 decodes Type 0 Configuration transactions received on its NT Port Link Interface. The PEX 8619 reads from or writes to the NT Port Link Interface register, as specified in the original Type 0 Configuration access.

#### 17.3.1.1 [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 17-1](#).) The *PCI r3.0-Compatible Configuration* mechanism provides standard access to the PEX 8619 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](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

### 17.3.1.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism 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)
- NT Port Link Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)
- NT Port Link Interface Power Budget Extended Capability Registers (Offsets 138h – 144h)
- NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)
- NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)
- 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)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 17.3.2 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 128-KB Memory map, as illustrated in [Figure 17-1](#). This Memory map is identical for Upstream Port **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. In NT PCI-to-PCI Bridge mode (**STRAP\_NT\_P2P\_EN#=L**), 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 **Debug Control** register *NT Port Number* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **1DCh[27:24]**)).

To use this mechanism, use the [PCI r3.0-Compatible Configuration Mechanism](#) to program the PEX 8619 upstream Port **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, offsets **10h** and **14h**, respectively). After the PEX 8619 NT Port Link Interface Memory-Mapped register Base address is Set, the PEX 8619 Configuration Space registers are accessed, using Memory Reads and Writes to the 4-KB range, starting at offset 64 KB (**1\_0000h**, Virtual Interface) and offset 68 KB (**1\_1000h**, Link Interface).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Figure 17-1. NT Mode Configuration Register Mapping to Memory-Mapped BAR**

PEX 8619

Port 0	0 KB: 0_0000h
Port 1	4 KB: 0_1000h
Port 2	8 KB: 0_2000h
Port 3	12 KB: 0_3000h
Port 4	16 KB: 0_4000h
Port 5	20 KB: 0_5000h
Port 6	24 KB: 0_6000h
Port 7	28 KB: 0_7000h
Port 8	32 KB: 0_8000h
Port 9	36 KB: 0_9000h
Port 10	40 KB: 0_A000h
Port 11	44 KB: 0_B000h
Port 12	48 KB: 0_C000h
Port 13	52 KB: 0_D000h
Port 14	56 KB: 0_E000h
Port 15	60 KB: 0_F000h
NT Port Virtual Interface	64 KB: 1_0000h
NT Port Link Interface	68 KB: 1_1000h
Reserved	
128 KB: 2_0000h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 17.3.3 Device-Specific Cursor Mechanism

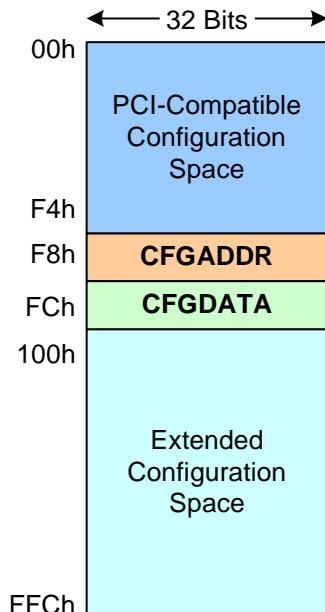
The Device-Specific Cursor mechanism is provided for use in development systems that can only generate *PCI r3.0* Configuration cycles (*that is*, the system cannot use either the [Device-Specific Memory-Mapped Configuration Mechanism](#), nor generate Extended Configuration Requests to access the Extended Configuration Space).

In [Figure 17-2](#), 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 17.10, “NT Port Link Interface Vendor-Specific Capability 3 Registers \(Offsets C8h – FCh\),”](#) for the register descriptions.

**Figure 17-2. Configuration Space View**



## 17.4 Register Descriptions

The remainder of this chapter details the PEX 8619 NT Port Link Interface registers, including:

- Bit/field names
- Description of register functions in the PEX 8619 NT Port Link and Virtual Interfaces
- Type (*such as* RW or HwInit; refer to [Table 14-5, “Register Types, Grouped by User Accessibility,”](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8619 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 17.5 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 17-2](#) defines the register map.

**Table 17-2. 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	
PCI Status		PCI Command	
PCI Class Code			PCI Revision ID
PCI BIST <i>(Not Supported)</i>	PCI Header Type	Master Latency Timer <i>(Not Supported)</i>	Cache Line Size
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		
Expansion ROM Base Address			
<i>Reserved</i>			Capability Pointer (40h)
<i>Reserved</i>			38h
Max_Lat ( <i>Reserved</i> )	Min_Gnt ( <i>Reserved</i> )	PCI Interrupt Pin	PCI Interrupt Line
			3Ch

### Register 17-1. 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 8619, if not overwritten by serial EEPROM or I <sup>2</sup> C.	RO	Yes	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-2. 04h PCI Command/Status**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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 8619 ignores Memory Space Requests received on the NT Port Link Interface 1 = PEX 8619 accepts Memory Space Requests received on the NT Port Link Interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8619 forwarding of Memory Requests upstream. Does not affect Message forwarding nor Completions.  0 = PEX 8619 handles Memory Requests received on the NT Port Virtual Interface as Unsupported Requests (UR); for Non-Posted Requests, the PEX 8619 returns a Completion with UR Completion status 1 = PEX 8619 forwards Memory Requests from the NT Port Virtual Interface to the Link Interface	RW	Yes	0
3	<b>Special Cycle Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</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 r2.0</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 -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## Register 17-2. 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 Link Interface –or– PEX_INTA# (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 r2.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r2.0</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 r2.0</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 8619 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> Always Cleared.	RsvdP	No	00b
27	<b>Signaled Target Abort</b> The NT Port Link Interface Sets this bit if either of the following conditions exist: <ul style="list-style-type: none"><li>NT Port Link Interface receives a Completion (from the Local Host) that has a Completion status of Completer Abort (CA), –or–</li><li>NT Port Link Interface receives a Memory-Mapped access with a Payload length greater than 1 DWord</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 updated, because the NT Port does not log the corresponding Requests that it forwards.	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## Register 17-2. 04h PCI Command/Status (Cont.)

Bit(s)	Description	Type	Serial EEPROM and i <sup>2</sup> C	Default
28	<b>Received Target Abort</b> <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 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 17-3. 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 8619. The PEX 8619 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>				
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 17-4. 0Ch Miscellaneous Control**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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 8619 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 r2.0</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-5. 10h Base Address 0  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> 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), to provide a 64-bit <b>BAR0/I</b> , by programming the <b>NT Port Link Interface BAR0/I Setup</b> register <b>BAR0/I Enable</b> field (NT Port Link Interface, offset E4h[1:0]) 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 8619 Port Configuration registers into Memory space. <i>Note:</i> 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8619 Port Configuration registers into Non-Prefetchable Memory space	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	0-0h
31:17	<b>Base Address 0</b> 128-KB-aligned Base address used for Memory-Mapped access to the 128-KB block of all PEX 8619 registers (4 KB, per Port).	RW	Yes	0-0h

**Register 17-6. 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
	RO when the <b>Base Address 0</b> register ( <b>BAR0</b> ) 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-7. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	0_000h
31:20	<b>Base Address 2</b> Resolution is 1 MB.	RW	Yes	000h

**Register 17-8. 1Ch Base Address 3**  
**(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> 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).				
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	RsvdP	No
		Offset 18h[2:1]=10b	RW	Yes
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	RsvdP	No
		Offset 18h[2:1]=10b	RO	Yes
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 18h[2:1]=00b	RsvdP	No
		Offset 18h[2:1]=10b	RW	Yes
19:4	<b>Reserved</b>	Offset 18h[2:1]=00b	RsvdP	No
	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
31:20	<b>Base Address 3</b>	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-9. 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RO	Yes	0
19:4	<b>Reserved</b>	RsvdP	No	0_000h
31:20	<b>Base Address 4</b>	RW	Yes	000h

**Register 17-10. 24h Base Address 5**  
**(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i>Note:</i> 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 <b>20h[2:1]</b> ), is programmed to 10b).					
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 Offset 20h[2:1]=10b	RsvdP RW	No Yes	0 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 Offset 20h[2:1]=10b	RsvdP RW	No Yes	00b 00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 20h[2:1]=00b Offset 20h[2:1]=10b	RsvdP RW	No Yes	0 0
19:4	<b>Reserved</b> 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]=00b Offset 20h[2:1]=10b	RsvdP RW	No Yes	0_000h 0_000h
31:20	<b>Base Address 5</b>	RW	Yes	000h	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-11. 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 or I <sup>2</sup> C.	RO	Yes	10B5h
<b>Subsystem ID</b>				
31:16	<b>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

**Register 17-12. 30h Expansion ROM Base Address**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Note:</i> 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 (<a href="#">Ingress Control</a> register <a href="#">Expansion ROM Virtual Side</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">660h[23]</a>) is Cleared).</p>				
0	<b>Expansion ROM Enable</b> 0 = NT Port Link Interface Expansion ROM is disabled	NT Port Virtual Interface, offset <a href="#">30h[0]=1</a>	RsvdP	No
	1 = NT Port Link Interface Expansion ROM is enabled, and NT Port Virtual Interface Expansion ROM is disabled	NT Port Virtual Interface, offset <a href="#">30h[0]=0</a>	RO	Yes
		NT Port Virtual Interface, offset <a href="#">30h[0]=1</a>	RsvdP	No
13:1	<b>Reserved</b>	RsvdP	No	0-0h
31:14	<b>Expansion ROM Base Address</b> If the <a href="#">Serial EEPROM Clock Frequency</a> register <a href="#">Expansion ROM Size</a> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">268h[16]</a> ) value is 0, the Expansion ROM size is 16 KB (default value is FFFF_C001h). Bit 14 is RW. If the <a href="#">Expansion ROM Size</a> bit value is 1, the Expansion ROM size is 32 KB (default value is FFFF_8001h). Bit 14 is RO.	When Bit 0 = 0  When Bit 0 = 1	RsvdP  RW	No  Yes
				0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-13. 34h Capability Pointer**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
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 17-14. 3Ch PCI Interrupt**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>PCI Interrupt Line</b>				
7:0	<b>Interrupt Line Routing Value</b> 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
<b>PCI Interrupt Pin</b>				
15:8	<b>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 8619. 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>Min_Gnt</b>				
23:16	<b>Minimum Grant</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h
<b>Max_Lat</b>				
31:24	<b>Maximum Latency</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 17.6 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 17-3](#) defines the register map.

**Table 17-3. 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	PCI Power Management Data	Next Capability Pointer (48h)	Capability ID (01h)	40h
PCI Power Management Control/Status Bridge Extensions ( <i>Reserved</i> )		PCI Power Management Status and Control		44h

### Register 17-15. 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	<a href="#">48h</a>
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 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 8619 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 8619 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8619 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.  <i>Note:</i> This field is <i>reserved</i> in NT PCI-to-PCI Bridge mode.	RO	Yes	0000_0b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-16. 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	<p><b>Power State</b>            Used to determine the current Device PM state of the Port, and to Set the Port into a new Device PM state.</p> <p>00b = D0            01b = D1 – <i>Not supported</i>            10b = D2 – <i>Not supported</i>            11b = D3hot</p> <p>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.</p>	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<p><b>No Soft Reset</b>            1 = Devices transitioning from the D3hot to D0 Device PM state, because of Power State commands, do not perform an internal reset</p>	RO	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<p><b>PME Enable</b>            Default value of 0 indicates that PME generation is disabled.</p> <p><i>Note:</i> This bit is reserved in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0
12:9	<p><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 <b>Data</b> and <b>Data Scale</b> registers (fields [31:24] and 14:13, respectively).</p> <p>0h = D0 power consumed            3h = D3hot power consumed            4h = D0 power dissipated            7h = D3hot power dissipated</p> <p>All other encodings are reserved.</p>	RO	Yes	0h
14:13	<p><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>Data</b> register value. The value and meaning of this field varies, depending upon which data value is selected by field [12:9] (<i>Data Select</i>). There are four internal <b>Data Scale</b> registers (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h.</p>	RO	Yes	00b
15	<p><b>PME Status</b>            0 = PME is not being generated by the NT Port</p> <p><i>Note:</i> This bit is reserved in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-16. 44h PCI Power Management Status and Control (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>PCI Power Management Control/Status Bridge Extensions</b>				
21:16	<b>Reserved</b>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <b>Reserved</b> 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 <b>Data Scale</b> value returned is 0h. Selected by field [12:9] ( <i>Data Select</i> ).	RO	Yes	00h

- a. With no serial EEPROM nor previous I<sup>2</sup>C programming, Reads return 00h for the **Data** and **Data Scale** registers (for all *Data Selects*).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.7 NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h)

The registers detailed in [Section 14.8, “MSI Capability Registers \(Offsets 48h – 64h\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-4](#) (register map), and [Register 17-17](#) through [Register 17-19](#).

**Table 17-4. NT Port Link 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 17-17. 48h MSI Capability**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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> output assertion are disabled	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = NT Port Link Interface can request only one Vector 001b through 111b = NT Port Link Interface can request two Vectors	RO	Yes	001b
22:20	<b>Multiple Message Enable</b> 000b = NT Port Link Interface is allocated one Vector, by default 001b through 111b = NT Port Link Interface is allocated two Vectors <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 8619 is capable of generating MSI 32-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the Message address) 1 = PEX 8619 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 8619 does not have Per Vector Masking capability 1 = PEX 8619 has Per Vector Masking capability	RO	Yes	1
31:25	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-18. 58h MSI Mask**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
The interrupt sources in the NT Port Link Interface are grouped into two categories – Device-Specific errors and NT-Link Doorbell-generated interrupts.					
The quantity of allocated MSI Vectors is determined by the <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). When the quantity of MSI Vectors that can be requested is:					
<ul style="list-style-type: none"> <li>• <b>Two</b> – Both interrupt categories generate their own MSI Vector</li> <li>• <b>One</b> – Both interrupt categories generate the same MSI Vector</li> </ul>					
<i>Notes:</i> 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.					
<i>The bits in this register can be used to mask their respective <b>MSI Status</b> register bits (offset 5Ch).</i>					
0	<b>MSI Mask for Device-Specific Errors</b> MSI mask for Device-Specific Error interrupts.	Offset 48h[22:20]≥001b	RW	Yes	0
	<b>MSI Mask for Shared Interrupt Sources</b> MSI mask for both interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one Vector.	Offset 48h[22:20]=000b	RsvdP	No	0
1	<b>MSI Mask for NT-Link Doorbell-Generated Interrupts</b> Refer to the 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
31:2	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-19. 5Ch MSI Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
--------	-------------	------	------------------------------------	---------

The interrupt sources in the NT Port Link Interface are grouped into two categories – Device-Specific errors and NT-Link Doorbell-generated interrupts.

The quantity of allocated MSI Vectors is determined by the **MSI Control** register *Multiple Message Capable* and *Multiple Message Enable* fields (offset 48h[19:17] and 22:20], respectively). When the quantity of MSI Vectors that can be requested is:

- Two – Both interrupt categories generate their own MSI Vector
- One – Both interrupt categories generate the same MSI Vector

*Notes:* The offset for this register changes from 5Ch, to 58h, when the **MSI Control** register *MSI 64-Bit Address Capable* bit (offset 48h[23]) is Cleared.

The bits in this register can be masked by their respective **MSI Mask** register bits (offset 58h).

0	<b>MSI Pending Status for Device-Specific Errors</b> MSI pending status for Device-Specific Error interrupts.	Offset 48h[22:20]≥001b	RO	Yes	0
	<b>MSI Pending Status for Shared Interrupt Sources</b> MSI pending status for both interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one Vector.	Offset 48h[22:20]=000b	RsvdP	No	0
1	<b>MSI Pending Status for NT-Link Doorbell-Generated Interrupts</b> Refer to the NT Port registers located at offsets C5Ch through C68h.	Offset 48h[22:20]≥001b	RO	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0
31:2	<b>Reserved</b>		RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.8 NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)

The registers detailed in [Section 14.9, “PCI Express Capability Registers \(Offsets 68h – A0h\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-5](#) (register map; offsets 7Ch, 80h, and 90h are *reserved*), and [Register 17-20](#) through [Register 17-25](#).

**Table 17-5. 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 (A4h)	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 – 94h
Link Status 2		Link Control 2			98h	
<i>Reserved</i>						9Ch – A0h

**Register 17-20. 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 <a href="#">A4h</a> , to point to the <b>Subsystem Capability</b> structure.	RO	Yes	<a href="#">A4h</a>
<b>PCI Express Capability</b>				
19:16	<b>Capability Version</b> The PEX 8619 NT Port Link Interface programs this field to 2h, as required by the <i>PCI Express Base r2.0</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> <i>Note:</i> This bit is also <i>reserved</i> in NT PCI-to-PCI Bridge mode.	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-21. 6Ch Device Capability**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
2:0	<p><b>Maximum Payload Size Supported</b>            Maximum Payload Size Port limitations are as follows:            If there are no x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 2,048 bytes. If there are x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 512 bytes.</p> <p>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.</p>	HwInit	Yes	100b
4:3	<p><b>Phantom Functions Supported</b>  <i>Not supported</i></p>	RO	Yes	00b
5	<p><b>Extended Tag Field Supported</b>            0 = Maximum Tag field is 5 bits            1 = Maximum Tag field is 8 bits</p>	RO	Yes	0
8:6	<p><b>Endpoint L0s Acceptable Latency</b>            111b = No Limit</p>	RO	Yes	111b
11:9	<p><b>Endpoint L1 Acceptable Latency</b>            111b = No Limit</p>	RO	Yes	111b
14:12	<b>Reserved</b>	RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>	RO	Yes	1
17:16	<b>Reserved</b>	RsvdP	No	00b
25:18	<p><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>).</p>	RO	Yes	00h
27:26	<p><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</p>	RO	Yes	00b
31:28	<b>Reserved</b>	RsvdP	No	0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-22. 70h Device Status and Control**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<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 System 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 System 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 System 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>	RsvdP	No	0
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.  <i>Note:</i> Software must halt all transactions through the NT Port before changing this field.	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>	RsvdP	No	0
14:12	<b>Maximum Read Request Size</b> <i>Not supported</i>	RsvdP	No	000b
15	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-22. 70h Device Status and Control (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> Set when the NT Port Link Interface detects a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state. 0 = NT Port Link Interface did not detect a Correctable error 1 = NT Port Link Interface detected a Correctable error	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> Set when the NT Port Link Interface detects a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state. 0 = NT Port Link Interface did not detect a Non-Fatal error 1 = NT Port Link Interface detected a Non-Fatal error	RW1C	Yes	0
18	<b>Fatal Error Detected</b> Set when the NT Port Link Interface detects a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state. 0 = NT Port Link Interface did not detect a Fatal error 1 = NT Port Link Interface detected a Fatal error	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> Set when the NT Port Link Interface detects a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state. 0 = NT Port Link Interface did not detect a UR 1 = NT Port Link Interface detected a UR	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 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## Register 17-23. 74h Link Capability

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Supported Link Speeds</b>            Indicates the NT Port Link Interface's supported Link speed.            0001b = 2.5 GT/s Link speed is supported            0010b = 5.0 GT/s and 2.5 GT/s Link speeds are supported            All other encodings are <i>reserved</i>.</p>	RO	Yes	0010b (STRAP_RESERVED17#=H) 0001b (STRAP_RESERVED17#=L)
9:4	<p><b>Maximum Link Width</b>            The PEX 8619 maximum Link width is x8 = 00_1000b. Valid widths are x1, x4, or x8. Actual maximum Link width is Set by the <a href="#">STRAP_PORTCFG[3:0]</a> inputs.            00_0000b = <i>Reserved</i>            00_0001b = x1            00_0010b = x2            00_0100b = x4            00_1000b = x8            All other encodings are <i>not supported</i>.</p>	ROS	No	Set by STRAP_PORTCFG[3:0] input levels, or by serial EEPROM value for the <a href="#">Port Configuration</a> register <i>Port Configuration</i> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">574h[3:0]</a> )
11:10	<p><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>.</p>	RO	Yes	11b
14:12	<p><b>L0s Exit Latency</b>            Indicates the L0s Link PM state exit latency for the given PCI Express Link. Value depends upon the <a href="#">Common Physical Layer Command/Status</a> register <i>N_FTS Value</i> field (NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <a href="#">238h[15:8]</a>) value and Link speed. Exit latency is calculated, as follows:  <ul style="list-style-type: none"> <li>• <b>2.5 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 N_FTS) x 4 ns (1 symbol time at 2.5 GT/s)</li> <li>• <b>5.0 GHz</b> – Multiply <i>N_FTS Value</i> x 4 (4 symbol times in 1 N_FTS) x 2 ns (1 symbol time at 5.0 GT/s)</li> </ul>           100b = NT Port Link Interface L0s Link PM state Exit Latency is 512 ns to less than 1 µs at 5.0 GT/s            101b = NT Port Link Interface L0s Link PM state Exit Latency is 1 µs to less than 2 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.</p>	RO	No	100b (5.0 GT/s) 101b (2.5 GT/s)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-23. 74h Link Capability (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
17:15	<p><b>L1 Exit Latency</b>            Indicates the L1 Link PM state exit latency for the given PCI Express Link. Value depends upon the Link speed.</p> <p>001b = NT Port Link Interface L1 Link PM state Exit Latency is 1 µs to less than 2 µs at 5.0 GT/s            010b = NT Port Link Interface L1 Link PM state Exit Latency is 2 µs to less than 4 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.</p>	RO	Yes	001b (5.0 GT/s) 010b (2.5 GT/s)
18	<b>Clock Power Management</b>	RO	Yes	0
23:19	<p><b>Reserved</b></p> <p><i>Note:</i> Bits [21:19] are also reserved in NT PCI-to-PCI Bridge mode.</p>	RsvdP	No	0-0h
31:24	<p><b>Port Number</b>            The NT Port Number value is selected by the <b>STRAP_NT_UPSTRM_PORTSEL[3:0]</b> inputs.            All other encodings are <i>reserved</i>.</p> <p>00h = Port 0            01h = Port 1            02h = Port 2            03h = Port 3            04h = Port 4            05h = Port 5            06h = Port 6            07h = Port 7            08h = Port 8            09h = Port 9            0Ah = Port 10            0Bh = Port 11            0Ch = Port 12            0Dh = Port 13            0Eh = Port 14            0Fh = Port 15</p>	ROS	No	Set by STRAP_NT_UPSTRM_PORTSEL[3:0] input levels

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-24. 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.	RsvdP	No	0
5	<b>Retrain Link</b> <i>Reserved</i> for the NT Port Link Interface. Always read as 0.	RsvdP	No	0
6	<b>Common Clock Configuration</b> 0 = NT Port Link Interface and the device at the other end of the corresponding Port’s PCI Express Link use an asynchronous Reference Clock source 1 = NT Port Link Interface and the device at the other end of the corresponding Port’s PCI Express Link use a common (synchronous) Reference Clock source (constant phase relationship)	RW	Yes	0
7	<b>Extended Sync</b> When Set, 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 <i>Recovery</i> state.</li></ul>	RW	Yes	0
15:8	<b>Reserved</b>	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-24. 78h Link Status and Control (Cont.)**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<b>Link Status</b>				
19:16	<p><b>Current Link Speed</b>            Indicates the negotiated Link speed of the Port's PCI Express Link.            0001b = 2.5 GT/s Link speed            0010b = 5.0 GT/s Link speed            All other encodings are <i>reserved</i>. The value in this field is undefined when the Link is not up.</p>	RO	No	0001b
25:20	<p><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.            00_0000b = Link is down (default)            00_0001b = x1 or Port is in the <i>DL_Down</i> state            00_0010b = x2            00_0100b = x4            00_1000b = x8            All other encodings are <i>not supported</i>.</p>	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	<p><b>Slot Clock Configuration</b>            Set by the upstream Port or NT Port Link Interface, but not both.            0 = Indicates that the PEX 8619 uses an independent clock            1 = Indicates that the PEX 8619 uses the same physical Reference Clock that the platform provides on the connector</p>	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.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 17-25. 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> 0001b = 2.5 GT/s Link speed supported 0010b = 5.0 GT/s Link speed supported All other encodings are <i>reserved</i> .	RWS	Yes	0010b
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> 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
12	<b>Compliance De-Emphasis</b> Sets the de-emphasis level in the <i>Polling.Compliance</i> state, if the entry occurred due to bit 4 ( <a href="#">Enter Compliance</a> ) being Set.	RWS	Yes	0
15:13	<b>Reserved</b>	RsvdP	No	000b
<b>Link Status 2</b>				
16	<b>Current De-Emphasis Level</b> Reflects the de-emphasis level. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at 2.5 GT/s)	RO	No	0 (5.0 GT/s) 1 (2.5 GT/s)
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 17.9 NT Port Link Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)

The registers detailed in [Section 14.10, “Subsystem ID and Subsystem Vendor ID Capability Registers \(Offsets A4h – FCh\),”](#) are also applicable to the NT Port, except as defined in [Table 17-6](#) (register map) and [Register 17-26](#).

**Table 17-6. NT Port Link Interface Subsystem ID and Subsystem Vendor ID 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		
<i>Reserved</i>	Next Capability Pointer (C8h)	SSID/SSVID Capability ID (0Dh)	A4h
Subsystem ID	Subsystem Vendor ID		A8h
	<i>Reserved</i>		ACh – C4h

### Register 17-26. A4h Subsystem Capability

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>SSID/SSVID Capability ID</b> Detects the 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> Program to C8h, to point to the <b>Vendor-Specific Capability 3</b> structure.	RO	Yes	C8h
31:16	<i>Reserved</i>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.10 NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)

This section details the NT Port Link Interface Vendor-Specific Extended Capability 3 registers. [Table 17-7](#) defines the register map used by the NT Port.

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 17-7. NT Port Link Interface Vendor-Specific Extended 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		
Vendor-Specific Extended Capability 3	Next Capability Pointer 3 (00h)	PCI Express Capability ID 3 (09h)	C8h
<i>Vendor-Specific Header 3 (Reserved)</i>			CCh
<i>Reserved</i>			D0h – E0h
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	<i>Reserved</i>		F8h
Configuration Data Window			FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-27. C8h Vendor-Specific Extended Capability 3**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
7:0	<b>PCI Express Capability ID 3</b>	RO	Yes	09h
15:8	<b>Next Capability Pointer 3</b> 00h = This capability is the last capability in the Linked list	RO	Yes	00h
31:16	<b>Length</b> Quantity of bytes in this Capability structure.	RO	Yes	0038h

**Register 17-28. CCh Vendor-Specific Header 3**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
31:0	<b>Reserved</b>	RO	Yes	0380_0002h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-29. 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 = <b>Reserved</b> 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)	RW	No	10b
2	<b>BAR0 Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	No	0
31:3	<b>Reserved</b>	RsvdP	No	0-0h

**Register 17-30. 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.	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	Yes	0
19:4	<b>Reserved</b>	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 0, and Writes are ignored 1 = Corresponding <b>BAR2</b> bits are RW bits	RW	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	RW	Yes
	<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	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-31. ECh NT Port Link Interface Memory BAR2/3 Setup**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> This register has RW privilege if <b>BAR2/3</b> is configured as a 64-bit BAR ( <a href="#">NT Port Link Interface Memory BAR2 Setup</a> register <b>BAR2 Type</b> field, offset <a href="#">E8h[2:1]</a> , is programmed to 10b).				
0	<b>Type Selector</b>	Offset E8h[2:1]=00b	RsvdP	No
		Offset E8h[2:1]=10b	RW	Yes
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
		Offset E8h[2:1]=10b	RW	Yes
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset E8h[2:1]=00b	RsvdP	No
		Offset E8h[2:1]=10b	RW	Yes
19:4	<b>Reserved</b>	Offset E8h[2:1]=00b	RsvdP	No
	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	RW	Yes
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 0, and Writes are ignored 1 = Corresponding <b>BAR3</b> bits are RW bits		RW	Yes
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	RW	Yes
	<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	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-32. F0h NT Port Link Interface Memory BAR4 Setup**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
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.	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	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 0, and Writes are ignored 1 = Corresponding <b>BAR4</b> bits are RW bits	RW	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	RW	Yes
	<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	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-33. F4h NT Port Link Interface Memory BAR4/5 Setup**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
<i>Note:</i> This register has RW privilege if <b>BAR4/5</b> is configured as a 64-bit BAR ( <b>NT Port Link Interface Memory BAR4 Setup</b> register <b>BAR4 Type</b> field, offset <b>F0h[2:1]</b> , is programmed to 10b).				
0	<b>Type Selector</b>	Offset F0h[2:1]=00b	RsvdP	No
		Offset F0h[2:1]=10b	RW	Yes
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
		Offset F0h[2:1]=10b	RW	Yes
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset F0h[2:1]=00b	RsvdP	No
		Offset F0h[2:1]=10b	RW	Yes
19:4	<b>Reserved</b> 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]=00b	RsvdP	No
		Offset F0h[2:1]=10b	RW	Yes
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 0, and Writes are ignored 1 = Corresponding <b>BAR5</b> bits are RW bits		RW	Yes
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	RW	Yes
	<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	RW	Yes

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-34. F8h Configuration Address Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Reserved</b>	RsvdP	No	0000h
25:16	<b>Register Offset</b>	RW	Yes	0-0h
30:26	<b>Reserved</b>	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 17-35. 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 17.11 NT Port Link Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)

The registers detailed in [Section 14.11, “Device Serial Number Extended Capability Registers \(Offsets 100h – 134h\),”](#) are also applicable to the NT Port. [Table 17-8](#) defines the register map used by all Ports.

**Table 17-8. NT Port Link Interface Device Serial Number 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 (FB4h)	Capability Version (1h)	PCI Express Extended Capability ID (0003h)
Serial Number (Lower DW)		100h
Serial Number (Upper DW)		104h
<i>Reserved</i>		108h
		10Ch – 134h

## 17.12 NT Port Link Interface Power Budget Extended Capability Registers (Offsets 138h – 144h)

The registers detailed in [Section 14.12, “Power Budget Extended Capability Registers \(Offsets 138h – 144h\),”](#) are also applicable to the NT Port Link Interface. [Table 17-9](#) defines the register map.

**Table 17-9. NT Port Link Interface Power Budget 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 (0004h)
<i>Reserved</i>		Data Select
Power Budget Data		138h
Power Budget Capability		13Ch
		140h
		144h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.13 NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)

The registers detailed in [Section 14.13, “Virtual Channel Extended Capability Registers \(Offsets 148h – 1BCh\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-10](#) (register map; offsets 1A8h through 1B4h are *reserved*) and [Register 17-36](#).

**Table 17-10. NT Port Link 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 (950h)	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
	VC1 Resource Capability	164h
	VC1 Resource Control	168h
VC1 Resource Status	<i>Reserved</i>	16Ch
	<i>Reserved</i>	170h – 1BCh

**Register 17-36. 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 r2.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</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 950h.	RO	No	950h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 17.14 NT Port Link Interface Device-Specific Registers (Offsets 1C0h – C88h)

The registers detailed in [Section 14.14, “Device-Specific Registers \(Offsets 1C0h – 51Ch\),”](#) and [Section 14.16, “Device-Specific Registers \(Offsets 530h – F8Ch\)”](#) (for offsets A00h through C88h), are unique to the PEX 8619 and not referenced in the *PCI Express Base r2.0*. These registers are also applicable to the NT Port Link Interface, except as defined in [Table 17-11](#) (register map; offsets 200h through 57Ch, 680h through 94Ch, and 964h through C30h, are **reserved**, and offset 960h is associated with an NT Port Link Interface-specific register) through [Table 17-16](#), and Register 17-37 through [Register 17-45](#).

Another NT Port Link Interface Device-Specific register is detailed in [Section 17.16, “NT Port Link Interface Device-Specific Registers – Source Queue Weight \(Offsets F00h – F3Ch\),”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 17-11. NT Port Link Interface Device-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	
		1C0h
		...
		1FCh
	<i>Reserved</i>	200h – 65Ch
		660h
		...
		67Ch
	<i>Reserved</i>	980h – 94Ch
Next Capability Offset 2 (C34h)	1h	PCI Express Extended Capability ID 2 (000Bh)
		950h
		...
		95Ch
		960h
	<i>Reserved</i>	964h – C30h
Next Capability Offset 4 (000h)	1h	PCI Express Extended Capability ID 4 (000Bh)
		C34h
		...
		C88h
		NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 17.14.1 NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets 1C0h – 1FCh)

The registers detailed in [Section 14.14.1, “Device-Specific Registers – Error Checking and Debug \(Offsets 1C0h – 1FCh\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-12](#) (register map; offsets 1C0h, 1C4h, 1CCh through 1DCh, and 1F4h are *reserved*), and [Register 17-37](#) through [Register 17-39](#).

**Table 17-12. 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	
	<b><i>Reserved</i></b>	1C0h – 1C4h
	<b>ECC Error Check Disable</b>	1C8h
	<b><i>Reserved</i></b>	1CCh – 1DCh
<b>Power Management Hot Plug User Configuration (Legacy NT Mode)</b>		1E0h
<b><i>Reserved</i> (NT PCI-to-PCI Bridge Mode)</b>		
<b>Egress Control and Status (Legacy NT Mode)</b>		1E4h
<b><i>Reserved</i> (NT PCI-to-PCI Bridge Mode)</b>		
<b>Bad TLP Counter</b>		1E8h
<b>Bad DLLP Counter</b>		1ECh
<b><i>Reserved</i></b>		1F0h – 1F4h
<b>ACK Transmission Latency Limit (Legacy NT Mode)</b>		1F8h
<b><i>Reserved</i> (NT PCI-to-PCI Bridge Mode)</b>		
<b><i>Factory Test Only</i></b>		1FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-37. 1C8h ECC Error Check Disable**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
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	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	RWS	Yes	0
2	<b>Software Force Error Enable</b> 1 = <b>Correctable Error Status</b> and <b>Uncorrectable Error Status</b> registers (offsets <b>FC4h</b> and <b>FB8h</b> , respectively) change from RW1CS to RW	RWS	Yes	0
3	<b>Software Force Non-Posted Request</b> 1 = Enables handling of errors associated with Posted TLPs as if those errors are associated with Non-Posted TLPs	RWS	Yes	0
4	<b>Reserved</b>	RsvdP	No	0
5	<b>Enable PEX_INTA# Interrupt Output(s) for Device-Specific Error-Triggered Interrupts</b> 0 = Device-Specific Error Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = Device-Specific Error Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)	RWS	Yes	0
6	<b>Enable PEX_INTA# Interrupt Output(s) for GPIO-Generated Interrupts</b> 0 = General-Purpose Input/Output (GPIO) Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = GPIO Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)	RWS	Yes	0
7	<b>Enable PEX_INTA# Interrupt Output(s) for NT-Link Doorbell-Generated Interrupts</b> 0 = NT Port Link Interface Doorbell Interrupt Requests send an INTx Message (and do not assert PEX_INTA#) 1 = NT Port Link Interface Doorbell Interrupt Requests assert PEX_INTA# (and do not send an INTx Message)	RWS	Yes	0
8	<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:</i> This bit must remain Cleared, for compliance to specifications governing the MSI Capability.	RWS	Yes	0
31:9	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-38. 1E0h Power Management Hot Plug User Configuration**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> This register is reserved in NT PCI-to-PCI Bridge mode.				
0	<b>L0s Entry Idle Counter</b> Traffic idle time to meet to enter the L0s Link PM state. 0 = Idle condition must last 1 μs 1 = Idle condition must last 4 μs	RW	Yes	0
1	<b>Factory Test Only</b>	RW	Yes	0-0h
2	<b>NT Virtual MPS CSR Select</b> NT Port Virtual Interface Maximum Payload Size register select. 0 = NT Port Virtual Interface uses the NT Port Link Interface's <a href="#">Maximum Payload Size</a> (offset 70h[7:5]) 1 = NT Port Virtual Interface uses its own <a href="#">Maximum Payload Size</a> (offset 70h[7:5])	RW	Yes	0
7:3	<b>Factory Test Only</b>	RW	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 μs. 0 = Enables Link retraining when no DLLPs are received for more than 256 μs (default) 1 = DLLP Timeout is disabled	RW	Yes	0
9	<b>Factory Test Only</b>	RW	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	RW	Yes	0
31:11	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-39. 1F8h ACK Transmission Latency Limit**

<b>Bit(s)</b>	<b>Description</b>	<b>Type</b>	<b>Serial EEPROM and I<sup>2</sup>C</b>	<b>Default</b>
The value of this register should be valid after Link negotiation.				
<i>Note:</i> This register is <b>reserved</b> in NT PCI-to-PCI Bridge mode.				
11:0	<b>ACK Transmission Latency Limit</b> Acknowledge Control Packet (ACK) Transmission Latency Limit. The value of this field changes, based upon the <a href="#">Negotiated Link Width</a> (offset 78h[25:20]), after the Link is up. x1 Link width = 255d x2 Link width = 217d x4 Link width = 118d x8 Link width = 107d	RWS	Yes	0FFh
15:12	<b>Reserved</b>	RsvdP	No	0h
23:16	<b>Upper 8 Bits of the Replay Timer Limit</b> The value in this register is a multiplier of the default internal timer values that are compliant to the <i>PCI Express Base r2.0</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.14.2 NT Port Link Interface Device-Specific Registers – Ingress Control and Port Enable (Offsets 660h – 67Ch)

The registers detailed in [Section 14.16.5, “Device-Specific Registers – Ingress Control and Port Enable \(Offsets 660h – 67Ch\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-13](#) (register map; offsets 660h, 664h, and 674h through 678h are *reserved*).

**Table 17-13. NT Port Link Interface Device-Specific Ingress Control and Port Enable 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>	660h –	664h
<i>Reserved</i>	Port Enable Status		668h
Negotiated Link Width for Ports 0, 1, 2, 3, 4, 5, 6, 7			66Ch
Negotiated Link Width for Ports 8, 9, 10, 11, 12, 13, 14, 15			670h
	<i>Reserved</i>	674h –	67Ch

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

### 17.14.3 NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets 950h – 95Ch)

The registers detailed in [Section 14.16.10, “Device-Specific Registers – Vendor-Specific Extended Capability 2 \(Offsets 950h – 95Ch\),”](#) are also applicable to the NT Port, except for the Next Capability Offset value, as defined in [Table 17-14](#) (register map) and [Register 17-40](#).

**Table 17-14. NT Port Link Interface Device-Specific, Vendor-Specific Extended Capability 2 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 2 (1h)	PCI Express Extended Capability ID 2 (000Bh)	950h
Vendor-Specific Header 2			954h
Hardwired Device ID		Hardwired Vendor ID	958h
<i>Reserved</i>			95Ch

**Register 17-40. 950h Vendor-Specific Extended Capability 2**

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, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	RO	Yes	000Bh
19:16	<b>Capability Version 2</b>	RO	Yes	1h
31:20	<b>Next Capability Offset 2</b> Program to C34h, which addresses the <b>Vendor-Specific Extended Capability 4</b> structure.	RO	Yes	C34h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 17.14.4 NT Port Link Interface Device-Specific Registers – Ingress NT Control (Offset 960h)

This section details the Device-Specific Ingress NT Control register, which is specific to the NT Port Link Interface. [Table 17-15](#) defines the register map.

**Table 17-15. NT Port Link Interface Ingress NT Control 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	Ingress NT Control 0	960h
-------------------------	-------------------------	-----------------------	-----------------	----------------------	------

#### Register 17-41. 960h Ingress NT Control 0

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> Use of this register is described in <a href="#">Section 15.4, “Traffic Class Translation.”</a>				
0	<b>TC Change Enable</b> Traffic Class Change enable. 0 = No change 1 = Perform TC change on Egress TLPs	RW	Yes	0
1	<b>BAR2_32_bit_ok</b>	RW	Yes	0
2	<b>BAR3_32_bit_ok</b>	RW	Yes	0
3	<b>BAR4_32_bit_ok</b>	RW	Yes	0
4	<b>BAR5_32_bit_ok</b>	RW	Yes	0
5	<b>BAR2_64_bit_ok</b>	RW	Yes	0
6	<b>BAR4_64_bit_ok</b>	RW	Yes	0
9:7	<b>Link to Virtual TC</b>	RW	Yes	000b
31:10	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 17.14.5 NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)

The registers detailed in Section 16.13.4, “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 17-16](#) (register map), and [Register 17-42](#) through [Register 17-45](#).

**Table 17-16. 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 (000h)	Capability Version 4 (1h)	PCI Express Extended Capability ID 4 (000Bh)
		Vendor-Specific Header 4
		Memory BAR2 Address Translation Lower
		Memory BAR3 Address Translation Upper
		Memory BAR4 Address Translation Lower
		Memory BAR5 Address Translation Upper
<i>Reserved</i>		Virtual Interface IRQ Set
<i>Reserved</i>		Virtual Interface IRQ Clear
<i>Reserved</i>		Virtual Interface IRQ Mask Set
<i>Reserved</i>		Virtual Interface IRQ Mask Clear
<i>Reserved</i>		Link Interface IRQ Set
<i>Reserved</i>		Link Interface IRQ Clear
<i>Reserved</i>		Link Interface IRQ Mask Set
<i>Reserved</i>		Link Interface IRQ Mask Clear
		NT Port SCRATCH0
		NT Port SCRATCH1
		NT Port SCRATCH2
		NT Port SCRATCH3
		NT Port SCRATCH4
		NT Port SCRATCH5
		NT Port SCRATCH6
		NT Port SCRATCH7

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 17-42. 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 ( <a href="#">NT Port Link Interface Memory BAR2 Setup</a> register <b>BAR2 Enable</b> bit, offset E8h[31], is Set).	RW	Yes	000h

**Register 17-43. 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 ECh[31]=0	RsvdP	No
	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 ECh[31]=1	RW	Yes
31:20	<b>NT Port Link-to-Virtual Interface BAR3 Base Translation Address</b> Base Translation address when <b>BAR3</b> is enabled ( <a href="#">NT Port Link Interface Memory BAR2/3 Setup</a> register <b>BAR3 Enable</b> bit, offset ECh[31], is Set).	RW	Yes	000h

**Register 17-44. 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 ( <a href="#">NT Port Link Interface Memory BAR4 Setup</a> register <b>BAR4 Enable</b> bit, offset F0h[31], is Set).	RW	Yes	000h

**Register 17-45. 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 F4h[31]=0	RsvdP	No
	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 F4h[31]=1	RW	Yes
31:20	<b>NT Port Link-to-Virtual Interface BAR5 Base Translation Address</b> Base Translation address when <b>BAR5</b> is enabled ( <a href="#">NT Port Link Interface Memory BAR4/5 Setup</a> register <b>BAR5 Enable</b> bit, offset F4h[31], is Set).	RW	Yes	000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 17.15 NT Port Link Interface NT Bridging-Specific Registers (Offsets C8Ch – EFCh)

[Table 17-17](#) defines the register map of the NT Port Link Interface NT Bridging-Specific registers.

**Table 17-17. 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	
	<i>Reserved</i>	C8Ch – DB0h
NT Port Link Interface Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h)		DB4h ... DF0h
	<i>Reserved</i>	DF4h – EFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 17.15.1 NT Port Link Interface 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 17-18 defines the register and address locations, as they relate to Register 17-46.

**Table 17-18. 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
DBCCh	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 17-46. 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_n Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
1	<b>LUT Entry_n 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>Reserved</b>		RsvdP	No	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_m Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
17	<b>LUT Entry_m 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>Reserved</b>		RsvdP	No	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 17.16 NT Port Link Interface Device-Specific Registers – Source Queue Weight (Offsets F00h – F3Ch)

Register offset F10h, detailed in [Section 14.16.17, “Device-Specific Registers – Source Queue Weight and Soft Error \(Offsets F00h – F3Ch\),”](#) is also applicable to the NT Port Link Interface. The remaining registers within the register set are *reserved* in the NT Port Link Interface, as defined in the [Table 17-19](#) register map.

Other NT Port Link Interface Device-Specific registers are detailed in [Section 17.14, “NT Port Link Interface Device-Specific Registers \(Offsets 1C0h – C88h\).”](#)

**Table 17-19. NT Port Link Interface Device-Specific Source Queue Weight Register Map (Offsets F00h – F3Ch)**

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	F00h –	F0Ch
		<i>Reserved</i>			
		Port Egress TLP Threshold			
		<i>Reserved</i>		F14h –	F3Ch

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 17.17 NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

The registers detailed in [Section 14.17, “Advanced Error Reporting Extended Capability Registers \(Offsets FB4h – FDCh\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 17-20 \(register map\)](#), [Register 17-47](#), and the exceptions indicated in the note.

**Notes:** *Data Link Protocol and Surprise Down PCI Express errors are not valid for the NT Port Link nor Virtual Interfaces, and therefore do not affect register offsets FB8h, FBCh, and FC0h in NT mode.*

*Receiver, Bad TLP, Bad DLLP, Replay Number Rollover, and Replay Timer Timeout PCI Express errors are not valid for the NT Port Link nor Virtual Interfaces, and therefore do not affect register offsets FC4h and FC8h in NT mode.*

**Table 17-20. 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
	Uncorrectable Error Status	FB8h
	Uncorrectable Error Mask	FBCh
	Uncorrectable Error Severity	FC0h
	Correctable Error Status	FC4h
	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 17-47. FB4h Advanced Error Reporting 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>	RO	Yes	0001h
19:16	<b>Capability Version</b>	RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to 138h, which addresses the upstream Port/NT Port Link Interface <b>Power Budget Extended Capability</b> structure.	RO	Yes	138h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3



## Chapter 18 DMA Interface Registers – DMA Mode Only

### 18.1 Introduction

This chapter defines the PEX 8619 DMA Interface registers when the PEX 8619 is operating in DMA mode. Other registers are defined in:

- Chapter 14, “Transparent Port Registers”
- Chapter 16, “NT Port Virtual Interface Registers – NT Mode Only”
- Chapter 17, “NT Port Link Interface Registers – NT Mode Only”

All PEX 8619 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 r2.0*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

1013

## 18.2 DMA Interface Type 0 Register Map

Table 18-1 defines the DMA Interface Type 0 register mapping.

**Table 18-1. DMA Interface Type 0 Register Map  
(DMA of Upstream 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	
		00h
		...
DMA Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)	Capability Pointer (40h)	34h
		...
	Next Capability Pointer (48h)      Capability ID (01h)	3Ch
DMA Interface PCI Power Management Capability Registers (Offsets 40h – 44h)		40h
	Next Capability Pointer (68h)      Capability ID (05h)	44h
DMA Interface MSI Capability Registers (Offsets 48h – 64h)		48h
	Next Capability Pointer (A4h)      Capability ID (10h)	...
DMA Interface PCI Express Capability Registers (Offsets 68h – A0h)		64h
	Next Capability Pointer (00h)      SSID/SSVID Capability ID (0Dh)	68h
DMA Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)		...
		A0h
DMA Interface Base Address DMA Setup Register (Offsets C8h – FCh)		A4h
		...
Next Capability Offset (FB4h)      1h      PCI Express Extended Capability ID (0003h)		C4h
DMA Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)		C8h
		...
Reserved		FCh
Next Capability Offset (000h)      1h      PCI Express Extended Capability ID (000Bh)		100h
		...
Vendor-Specific Extended Capability Registers – DMA Channel (Offsets 1F4h – 5FCh)		134h
		1F0h
		...
		1F4h
		...
		5FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 18-1. DMA Interface Type 0 Register Map  
(DMA of Upstream Port) (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	
		600h
DMA Device-Specific Registers – RAM Test and Soft Error (Offsets 600h – FB0h)		...
		FB0h
Next Capability Offset (1F4h)	1h	PCI Express Extended Capability ID (0001h)
		FB4h
DMA Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)		...
		FDCh
	<i>Reserved</i>	FE0h – FFCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.3 Register Access

The PEX 8619 DMA (Upstream Port, Function 1) 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 8619 supports two mechanisms for accessing DMA Interface registers:

- PCI Express Base r2.0 Configuration Mechanism
- Device-Specific Memory-Mapped Configuration Mechanism

### 18.3.1 PCI Express Base r2.0 Configuration Mechanism

The *PCI Express Base r2.0* 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 DMA Interface Configuration register space
- **PCI Express Enhanced Configuration Access Mechanism** – Provides access to the entire 4-KB Configuration Space

#### 18.3.1.1 PCI r3.0-Compatible Configuration Mechanism

*Do not use this mechanism to access the PEX 8619 Device-Specific Configuration registers.*

The *PCI r3.0*-Compatible Configuration mechanism provides standard access to the PEX 8619 DMA Interface's first 256 bytes (the bytes at offsets 00h through FFh) of the PCI Express Configuration Space.

This mechanism is used to access the following PEX 8619 DMA Interface Type 0 (PCI endpoint) registers:

- DMA Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)
- DMA Interface PCI Power Management Capability Registers (Offsets 40h – 44h)
- DMA Interface MSI Capability Registers (Offsets 48h – 64h)
- DMA Interface PCI Express Capability Registers (Offsets 68h – A0h)
- DMA Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)

The *PCI r3.0*-Compatible Configuration mechanism uses PCI Type 0 Configuration transactions to access the PEX 8619 Configuration registers.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

The PEX 8619 DMA Interface registers can be accessed from the upstream Port by Type 0 Configuration. A PEX 8619 DMA-enabled device decodes all Type 0 Configuration accesses received on its upstream Port for DMA Interface access, when any of the following conditions exist:

- If the specified Bus Number and Device Number that matches the upstream Port and Function Number specified in Configuration accesses is 1.
- If the specified Bus Number and Device Number that matches the upstream Port and Function Number specified in Configuration accesses is 1, but the register address is invalid, the PEX 8619 responds with an Unsupported Request (UR). The *Completer ID* field for Non-Posted accesses will match DMA's Bus Number, Device Number, and Function Number.
- If the specified Bus Number and Device Number that matches the upstream Port and Function Number specified in Configuration accesses is 1, but the *EP* bit is Set, the PEX 8619 responds with a UR. The *Completer ID* field for Non-Posted accesses will match DMA's Bus Number, Device Number, and Function Number.

Because the mechanism is limited to the first 256 bytes of the DMA Interface Configuration register space, the [PCI Express Enhanced Configuration Access Mechanism](#) or [Device-Specific Memory-Mapped Configuration Mechanism](#) must be used to access beyond Byte FFh.

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).

#### 18.3.1.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration mechanism 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.

This mechanism can be used to access all PEX 8619 Configuration registers.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 18.3.2 Device-Specific Memory-Mapped Configuration Mechanism

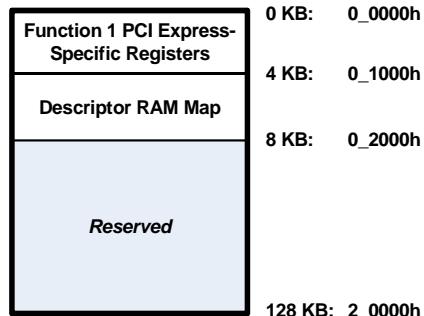
The Device-Specific Memory-Mapped Configuration mechanism provides a method to access the PEX 8619 DMA Interface Configuration Space, using Memory Read/Write transactions.

To use this mechanism, use the [PCI r3.0-Compatible Configuration Mechanism](#) to program the PEX 8619 DMA Interface **Base Address 0 DMA** and **Base Address 1 DMA** registers (**BAR0** and **BAR1**, offsets **10h** and **14h**, respectively). The DMA **BAR0/1** window size is 128 KB, out of which the first 4 KB is for PCI Express Configuration Space, and the next 4 KB, are mapped to Descriptor RAM, as illustrated in [Figure 18-1](#). Descriptor RAM is 256 bytes deep by 16 bytes wide, and can be initialized by Memory-Mapped access to **BAR0/1** at offsets 4 KB to 8 KB. This mapping allows users to randomly access each Descriptor.

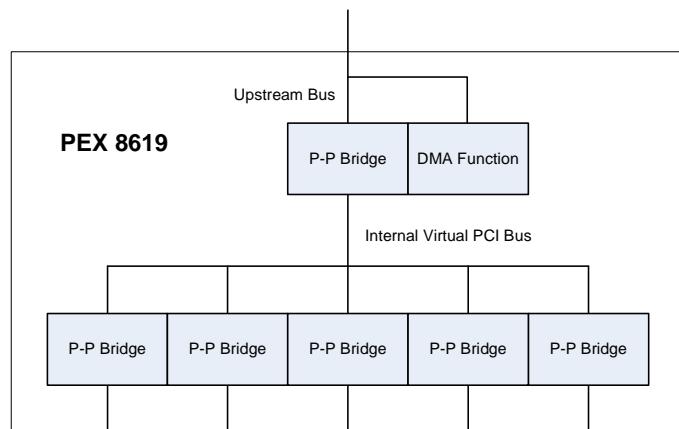
After the PEX 8619 DMA Interface Memory-Mapped register Base address is Set, the PEX 8619 Configuration Space/Device-Specific Space registers can be accessed, using Memory Reads and Writes.

*Note:* DMA Address space is not included within the upstream Port, although it is orthogonal to Upstream Port Address space, as illustrated in [Figure 18-2](#).

**Figure 18-1. DMA BAR0/1 Window**



**Figure 18-2. Software View, with DMA**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 18.4 Register Descriptions

The remainder of this chapter details the PEX 8619 DMA Interface registers, including:

- Bit/field names
- Description of register functions in DMA mode
- Type (*such as* RW or HwInit; refer to [Table 14-5, “Register Types, Grouped by User Accessibility,”](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8619 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

1019

## 18.5 DMA Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)

This section details the PCI-Compatible Type 0 Configuration Header registers in DMA mode. [Table 18-2](#) defines the register map.

**Table 18-2. DMA Interface PCI-Compatible Type 0 Configuration Header Register Map (DMA of Upstream 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																											
Device ID		Vendor ID																											
PCI Status		PCI Command																											
DMA PCI Class Code										DMA PCI Revision ID																			
PCI BIST <i>(Not Supported)</i>	PCI Header Type	Master Latency Timer <i>(Not Supported)</i>										Cache Line Size																	
Base Address 0 DMA																													
Base Address 1 DMA																													
<i>Reserved</i>																													
18h –		<i>Reserved</i>																											
Subsystem ID		Subsystem Vendor ID																											
Expansion ROM Base Address ( <i>Reserved</i> )																													
30h		<i>Reserved</i>																											
34h		Capability Pointer (40h)																											
<i>Reserved</i>																													
38h																													
3Ch		Max_Lat ( <i>Reserved</i> ) Min_Gnt ( <i>Reserved</i> ) PCI Interrupt Pin PCI Interrupt Line																											

**Register 18-1. 00h PCI Configuration ID (DMA of Upstream Port)**

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 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-2. 04h PCI Command/Status  
(DMA of Upstream Port)**

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 8619 ignores I/O Space accesses on the DMA Interface's primary interface 1 = PEX 8619 responds to I/O Space accesses on the DMA Interface's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8619 ignores Memory Space accesses on the DMA Interface's primary interface 1 = PEX 8619 responds to Memory Space accesses on the DMA Interface's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls the ability of the PEX 8619 DMA function to issue Memory Read/Write Requests. The DMA function does <i>not</i> support I/O Requests. Requests other than Memory are not controlled by this bit. 0 = DMA function is not allowed to issue Memory Requests 1 = DMA function is allowed to issue Memory Requests <i>Note:</i> MSI/MSI-X Interrupt Messages are in-band Memory Writes. Clearing this bit disables MSI/MSI-X Interrupt Messages as well.	RW	Yes	0
3	<b>Special Cycle Enable</b> <i>Not supported</i> Cleared, as required by the PCI Express Base r2.0.	RsvdP	No	0
4	<b>Memory Write and Invalidate Enable</b> <i>Not supported</i> Cleared, as required by the PCI Express Base r2.0.	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the PCI Express Base r2.0.	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 PCI Express Base r2.0.	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-2. 04h PCI Command/Status  
(DMA of Upstream Port) (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	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 r2.0</i> .	RsvdP	No	0
10	<b>Interrupt Disable</b> 0 = DMA Interface is enabled to generate INTx Interrupt Messages and assert <b>PEX_INTA#</b> output 1 = DMA Interface is prevented from generating INTx Interrupt Messages and asserting PEX_INTA# output	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-2. 04h PCI Command/Status  
(DMA of Upstream Port) (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 Message is pending 1 = INTx Interrupt Message is pending internally to the DMA Interface –or– PEX_INTA# (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 r2.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r2.0</i> .	RsvdP	No	0
22	<b>Reserved</b>	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 r2.0</i> .	RsvdP	No	0
24	<b>Master Data Parity Error Detected</b> If bit 6 ( <i>Parity Error Response Enable</i> ) is Set, the DMA Interface Sets this bit when the Requester poisons a Write Request or when the interface receives a Completion marked as poisoned. If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8619 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 r2.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> 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. 1 = DMA Function completed a Posted or Non-Posted Request as a Completer Abort error	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-2. 04h PCI Command/Status  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Received Target Abort</b> 1 = DMA Function received a Completion with Completer Abort Completion status	RW1C	Yes	0
29	<b>Received Master Abort</b> 1 = DMA Function received a Completion with Unsupported Request (UR) Completion status	RW1C	Yes	0
30	<b>Signaled System Error</b> If bit 8 ( <i>SERR# Enable</i> ) is Set, the DMA Interface 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 = DMA Interface received a Poisoned TLP, regardless of the bit 6 ( <i>Parity Error Response Enable</i> ) state	RW1C	Yes	0

**Register 18-3. 08h DMA PCI Class Code and Revision ID  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>DMA 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 8619. The PEX 8619 Serial EEPROM register Initialization capability is used to replace the PLX Revision ID with another Revision ID.	RO	Yes	BAh
<b>DMA PCI Class Code</b>				
15:8	<b>Register-Level Programming Interface</b> None defined.	RO	Yes	00h
23:16	<b>Sub-Class Code</b> Other bridge device.	RO	Yes	80h
31:24	<b>Base Class Code</b> Generic system peripherals.	RO	Yes	08h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-4. 0Ch DMA Miscellaneous Control  
(DMA of Upstream Port)**

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 8619 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 r2.0</i> .	RsvdP	No	00h
<b>PCI Header Type</b>				
22:16	<b>Configuration Layout Type</b> Type 0 Configuration Header used in DMA mode.	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, 64 DWords in size	RO	No	1
<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 -- kotlin-novator | vtuz vova

**Register 18-5. 10h Base Address 0 DMA  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> When enabled, the Base Address register maps PEX 8619 Port Configuration registers into Memory space. <b>BAR0 DMA</b> is configured by the serial EEPROM and Local Host. By default, the <b>BAR0/1 Setup</b> register selects 32-bit Memory <b>BAR0 DMA</b> . <i>Note:</i> 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> .	RO	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8619 Configuration registers into Non-Prefetchable Memory space	RO	Yes	0
16:4	<b>Reserved</b>	RsvdP	No	0-0h
31:17	<b>Base Address 0 DMA</b> 128-KB-aligned Base address used for Memory-Mapped access to the 128-KB block of all PEX 8619 registers (4 KB, per Port).	RW	Yes	0-0h

**Register 18-6. 14h Base Address 1 DMA  
(DMA of Upstream Port)**

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 DMA</b> ), Base Address 1 DMA ( <b>BAR1 DMA</b> ) extends Base Address 0 DMA ( <b>BAR0 DMA</b> ) to provide the upper 32 Address bits when the <b>Base Address 0 DMA</b> register <i>Memory Map Type</i> field (offset 10h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	RO when the <b>Base Address 0 DMA</b> register ( <b>BAR0 DMA</b> ) 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-7. 2Ch Subsystem ID and Subsystem Vendor ID  
(DMA of Upstream Port)**

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>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8619, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8619h

**Register 18-8. 30h Expansion ROM Base Address  
(DMA of Upstream Port)**

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 18-9. 34h Capability Pointer  
(DMA of Upstream Port)**

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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-10. 3Ch PCI Interrupt  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Interrupt Line</b>				
7:0	<b>Interrupt Line Routing Value</b> 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
<b>PCI Interrupt Pin</b>				
15:8	<b>Interrupt Pin</b> Identifies the Conventional PCI interrupt Message(s) the device (or device function) uses. In DMA mode, only values 00h, 01h, and 02h are allowed in the PEX 8619. 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	02h
<b>Min_Gnt</b>				
23:16	<b>Minimum Grant</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h
<b>Max_Lat</b>				
31:24	<b>Maximum Latency</b> <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 18.6 DMA Interface PCI Power Management Capability Registers (Offsets 40h – 44h)

This section details the PCI Power Management Capability registers in DMA mode. [Table 18-3](#) defines the register map.

**Table 18-3. DMA Interface PCI Power Management Capability Register Map (DMA of Upstream 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	
				40h
PCI Power Management Data	PCI Power Management Control/Status Bridge Extensions ( <i>Reserved</i> )	Next Capability Pointer (48h)	Capability ID (01h)	44h
		PCI Power Management Status and Control		

**Register 18-11. 40h PCI Power Management Capability (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID</b> Program to 01h, indicating 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>DMA MSI Capability</b> structure.	RO	Yes	48h
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 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 8619 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 8619 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8619 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-12. 44h PCI Power Management Status and Control  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Status and Control</b>				
1:0	<p><b>Power State</b> Used to determine the current Device PM state of the DMA Function, and to program the DMA Function into a new Device PM state.</p> <p>00b = D0 01b = D1 – <i>Not supported</i> 10b = D2 – <i>Not supported</i> 11b = D3hot</p> <p>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.</p>	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<p><b>No Soft Reset</b> 1 = Devices transitioning from the D3hot to D0 Device PM state, because of Power State commands, do not perform an internal reset</p>	RO	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<p><b>PME Enable</b> Default value of 0 indicates that PME generation is disabled.<sup>a</sup></p>	RsvdP	No	0
12:9	<p><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 <b>Data</b> and <b>Data Scale</b> registers (fields [31:24 and 14:13], respectively).</p> <p>0h = D0 power consumed 3h = D3hot power consumed 4h = D0 power dissipated 7h = D3hot power dissipated All other encodings are <i>reserved</i>.</p>	RO	Yes	0h
14:13	<p><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>Data</b> register value. The value and meaning of this field varies, depending upon which data value is selected by field [12:9] (<i>Data Select</i>). There are four internal <b>Data Scale</b> registers (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h.</p>	RO	Yes	00b
15	<p><b>PME Status</b> 0 = PME is not being generated in DMA mode<sup>a</sup></p>	RsvdP	No	0

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-12. 44h PCI Power Management Status and Control  
(DMA of Upstream Port) (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	<b>Reserved</b>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <b>Reserved</b> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <b>Reserved</b> 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). For other <i>Data Select</i> values, the <b>Data Scale</b> value returned is 0h. Selected by field [12:9] ( <i>Data Select</i> ).	RO	Yes	00h

- a. Because the PEX 8619 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 00h for the **Data** and **Data Scale** registers (for all Data Selects).

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.7 DMA Interface MSI Capability Registers (Offsets 48h – 64h)

This section details the Message Signaled Interrupt (MSI) Capability registers in DMA mode. [Table 18-4](#) defines the register map.

**Table 18-4. DMA Interface MSI Capability Register Map  
(DMA of Upstream Port)<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	
DMA MSI Control		Next Capability Pointer (68h)	Capability ID (05h)	48h
	DMA MSI Address			4Ch
	DMA MSI Upper Address			50h
Reserved		DMA MSI Data		54h
Reserved			DMA MSI Mask	58h
Reserved			DMA MSI Status	5Ch
	Reserved			60h – 64h

- a. Offsets 54h, 58h, and 5Ch change to 50h, 54h, and 58h, respectively, when the **DMA MSI Control** register **MSI 64-Bit Address Capable** bit (offset 48h[23]) is Cleared.

**Register 18-13. 48h DMA MSI Capability  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>DMA MSI Capability Header</b>				
7:0	<b>Capability ID</b> Program to 05h, as required by the PCI r3.0.	RO	Yes	05h
15:8	<b>Next Capability Pointer</b> Program to 68h, to point to the PCI Express Capability structure.	RO	Yes	68h
<b>DMA MSI Control</b>				
16	<b>MSI Enable</b> 0 = MSIs for the DMA Interface are disabled (Function 1 only) 1 = MSIs for the DMA Interface are enabled, and INTx Interrupt Messages and PEX_INTA# output assertion are disabled (Function 1 only)	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = PEX 8619 DMA Function can request only one Vector 001b = PEX 8619 DMA Function can request two Vectors 010b through 111b = PEX 8619 DMA Function can request four Vectors	RO	Yes	011b
22:20	<b>Multiple Message Enable</b> 000b = PEX 8619 DMA Function is allocated one Vector, by default 001b = PEX 8619 DMA Function is allocated two Vectors 010b through 111b = PEX 8619 DMA Function is allocated four Vectors <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 8619 is capable of generating MSI 32-bit addresses (DMA MSI Address register, offset 4Ch, is the Message address) 1 = PEX 8619 is capable of generating MSI 64-bit addresses (DMA MSI Address register, offset 4Ch, is the lower 32 bits of the Message address, and DMA MSI Upper Address register, offset 50h, is the upper 32 bits of the Message address)	RO	Yes	1
24	<b>Per Vector Masking Capable</b> 0 = PEX 8619 does not have Per Vector Masking capability 1 = PEX 8619 has Per Vector Masking capability	RO	Yes	1
31:25	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-14. 4Ch DMA MSI Address  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b>	RsvdP	No	00b
31:2	<b>Message Address</b> <i>Note:</i> Refer to register offset 50h for DMA MSI Upper Address, if offset 48h[23] is Set.	RW	Yes	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-15. 50h DMA MSI Upper Address  
(DMA of Upstream Port)**

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 <b>DMA 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:</i> Refer to register offset 4Ch for DMA MSI Address.	RW	Yes	0000_0000h

**Register 18-16. 54h DMA MSI Data  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The offset for this register changes from 54h, to 50h, when the <b>DMA MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Cleared.				
15:0	<b>Message Data</b> MSI Write transaction TLP payload.	RW	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-17. 58h DMA MSI Mask  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> The offset for this register changes from 58h, to 54h, when the <b>DMA MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Cleared.				
The bits in this register can be used to mask their respective <b>DMA MSI Status</b> register bits (offset 5Ch).				
7:0	<b>MSI Mask for Each Vector</b> The quantity of bits used depends upon the value of the <b>DMA MSI Control</b> register <b>Multiple Message Enable</b> field (offset 48h[19:17]). 00h = Bit 0 only 01h = Bits [1:0] 02h = Bits [3:0] 03h = Bits [7:0] All other encodings are <i>reserved</i> .	RW	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 18-18. 5Ch DMA MSI Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes:</i> The offset for this register changes from 5Ch, to 58h, when the <b>DMA MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Cleared.				
The bits in this register can be masked by their respective <b>DMA MSI Mask</b> register bits (offset 58h).				
7:0	<b>MSI Pending for Each Vector</b> The quantity of bits used depends upon the value of the <b>DMA MSI Control</b> register <b>Multiple Message Enable</b> field (offset 48h[19:17]). 00h = Bit 0 only 01h = Bits [1:0] 02h = Bits [3:0] 03h = Bits [7:0] All other encodings are <i>reserved</i> .	RO	No	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.8 DMA Interface PCI Express Capability Registers (Offsets 68h – A0h)

This section details the PCI Express Capability registers in DMA mode. [Table 18-5](#) defines the register map.

**Table 18-5. DMA Interface PCI Express Capability Register Map  
(DMA of Upstream 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			
PCI Express Capability		Next Capability Pointer (A4h)	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 – 8Ch
Device Status 2 (Reserved)		Device Control 2			90h	
Reserved						94h
Link Status 2		Link Control 2 (Reserved)			98h	
Reserved						9Ch – A0h

**Register 18-19. 68h PCI Express Capability List and Capability  
(DMA of Upstream Port)**

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 A4h, to point to the Subsystem Capability structure.	RO	Yes	A4h
<b>PCI Express Capability</b>				
19:16	<b>Capability Version</b> The PEX 8619 Ports programs this field to 2h, as required by the <i>PCI Express Base r2.0</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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-20. 6Ch Device Capability  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Maximum Payload Size Supported</b> Maximum Payload Size Port limitations are as follows: If there are no x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 2,048 bytes. If there are x1 Ports in the Port configuration, the Maximum Payload Size for each Port is 512 bytes. 000b = DMA Interface supports a 128-byte maximum payload 001b = DMA Interface supports a 256-byte maximum payload 010b = DMA Interface supports a 512-byte maximum payload 011b = DMA Interface supports a 1,024-byte maximum payload 100b = DMA Interface supports a 2,048-byte maximum payload No other encodings are supported.	HwInit	Yes	100b
4:3	<b>Phantom Functions Supported</b> <i>Not supported</i>	RO	Yes	00b
5	<b>Extended Tag Field Supported</b> 0 = Maximum Tag field is 5 bits 1 = Maximum Tag field is 8 bits	RO	Yes	1
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
17:16	<b>Reserved</b>	RsvdP	No	00b
25:18	<b>Captured Slot Power Limit Value</b> 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> 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-21. 70h Device Status and Control  
(DMA of Upstream Port)**

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 DMA Interface to report Correctable errors to the System Host	RW	Yes	0
1	<b>Non-Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the DMA Interface to report Non-Fatal errors to the System Host	RW	Yes	0
2	<b>Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the DMA Interface to report Fatal errors to the System Host	RW	Yes	0
3	<b>Unsupported Request Reporting Enable</b> 0 = Disables 1 = Enables the DMA Interface to report UR errors as error Messages with a programmed uncorrectable error severity	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> Relaxed ordering is supported in the DMA Function, on a per channel basis, in DMA-specific registers.	RW	Yes	0
7:5	<b>Maximum Payload Size</b> Software can change this field to configure the DMA 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 <b>Maximum Payload Size Supported</b> field (offset <b>6Ch[2:0]</b> ), for the DMA Interface.  000b = DMA Interface supports a 128-byte maximum payload 001b = DMA Interface supports a 256-byte maximum payload 010b = DMA Interface supports a 512-byte maximum payload 011b = DMA Interface supports a 1,024-byte maximum payload 100b = DMA Interface 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>	RW	Yes	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> No snoop is supported for the DMA Function, on a per channel basis.	RW	Yes	0
14:12	<b>Maximum Read Request Size</b>	RW	Yes	010b
15	<b>Reserved</b>	RsvdP	No	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-21. 70h Device Status and Control  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> Set when DMA Interface detects a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state. 0 = DMA Interface did not detect a Correctable error 1 = DMA Interface detected a Correctable error	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> Set when DMA Interface detects a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state. 0 = DMA Interface did not detect a Non-Fatal error 1 = DMA Interface detected a Non-Fatal error	RW1C	Yes	0
18	<b>Fatal Error Detected</b> Set when DMA Interface detects a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state. 0 = DMA Interface did not detect a Fatal error 1 = DMA Interface detected a Fatal error	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> Set when DMA Interface detects a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state. 0 = DMA Interface did not detect a UR 1 = DMA Interface detected a UR	RW1C	Yes	0
20	<b>AUX Power Detected</b> <i>Not supported</i>	RsvdP	No	0
21	<b>Transactions Pending</b> Cleared, as required by the <i>PCI Express Base r2.0</i> .	RO	No	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-22. 74h Link Capability  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Supported Link Speeds</b> Indicates the supported Link speed of the DMA Function, which has the same Link speed as the upstream Port. 0001b = 2.5 GT/s Link speed is supported 0010b = 5.0 GT/s and 2.5 GT/s Link speeds are supported All other encodings are <i>reserved</i>.</p>	RO	Yes	0010b (STRAP_RESERVED17#=H)  0001b (STRAP_RESERVED17#=L)
9:4	<p><b>Maximum Link Width</b> The PEX 8619 maximum Link width is x8 = 00_1000b. Valid widths are x1, x4, or x8. Actual maximum Link width is Set by the <b>STRAP_PORTCFG[3:0]</b> inputs. The DMA Function has the same maximum Link width as the upstream Port. 00_0000b = <i>Reserved</i> 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 All other encodings are <i>not supported</i>.</p>	ROS	No	Set by <b>STRAP_PORTCFG[3:0]</b> input levels, or by serial EEPROM <b>Port Configuration</b> register <b>Port Configuration</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <b>574h[3:0]</b> ) value
11:10	<p><b>Active State Power Management (ASPM) Support</b> Active State Link PM support. Indicates the level of ASPM supported by the DMA Function, which has the same ASPM encoding as the upstream 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>.</p>	RO	Yes	11b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-22. 74h Link Capability  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<p><b>L0s Exit Latency</b>            Indicates the L0s Link PM state exit latency for the given PCI Express Link. Value depends upon the <b>Common Physical Layer Command/Status</b> register <b>N_FTS Value</b> field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 238h[15:8]) value and Link speed. The DMA Function should have the same value as the upstream Port.            Exit latency is calculated, as follows:           <ul style="list-style-type: none"> <li>• <b>2.5 GHz</b> – Multiply <i>N_FTS Value</i> x 4            (4 symbol times in 1 N_FTS) x 4 ns            (1 symbol time at 2.5 GT/s)</li> <li>• <b>5.0 GHz</b> – Multiply <i>N_FTS Value</i> x 4            (4 symbol times in 1 N_FTS) x 2 ns            (1 symbol time at 5.0 GT/s)</li> </ul>           100b = DMA Interface's L0s Link PM state            Exit Latency is 512 ns to less than 1 µs at 5.0 GT/s            101b = DMA Interface's L0s Link PM state            Exit Latency is 1 µs to less than 2 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.         </p>	RO	No	100b (5.0 GT/s)  101b (2.5 GT/s)
17:15	<p><b>L1 Exit Latency</b>            Indicates the L1 Link PM state exit latency for the given PCI Express Link. Value depends upon the Link speed. The DMA Function should have the same L1 exit latency as the upstream Port.            001b = DMA Interface's L1 Link PM state            Exit Latency is 1 µs to less than 2 µs at 5.0 GT/s            010b = DMA Interface's L1 Link PM state            Exit Latency is 2 µs to less than 4 µs at 2.5 GT/s            All other encodings are <i>reserved</i>.         </p>	RO	Yes	001b (5.0 GT/s)  010b (2.5 GT/s)
18	<b>Clock Power Management</b>	RO	Yes	0
23:19	<b>Reserved</b>	RsvdP	No	0-0h
31:24	<p><b>Port Number</b>            The DMA Function has the same Port Number as the upstream Port.</p>	ROS	No	Default value same as Upstream Port Number

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Table 18-6. Port Configurations**

<b>STRAP_PORTCF G[3:0]</b>	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15
(default) 0h	x1	x1	x1	x1	x1	x1										
1h	x4	x1	x1	x1		x1		x1		x1	x1	x1	x1	x1	x1	x1
2h	x4	x4	x1	x1							x1	x1	x1	x1	x1	x1
3h	x4	x4	x4	x1							x1			x1		x1
4h	x4	x4	x4	x4												
5h	x8	x1		x1		x1		x1								
6h	x8	x4		x1								x1		x1		x1
7h	x8	x4		x4												
8h	x8	x8														

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-23. 78h Link Status and Control  
(DMA of Upstream Port)**

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 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> in DMA mode.	RsvdP	No	0
5	<b>Retrain Link</b> <i>Reserved</i> in DMA mode. Always read as 0.	RsvdP	No	0
6	<b>Common Clock Configuration</b> 0 = Upstream Port 0 and the device at the other end of Port 0’s PCI Express Link use an asynchronous Reference Clock source 1 = Upstream Port 0 and the device at the other end of Port 0’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 upstream Port 0 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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-23. 78h Link Status and Control  
(DMA of Upstream Port) (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 DMA Function's PCI Express Link. 0001b = 2.5 GT/s Link speed 0010b = 5.0 GT/s Link speed All other encodings are <i>reserved</i> . The value in this field is undefined when the Link is not up.	RO	No	0001b
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. 00_0000b = Link is down (default) 00_0001b = x1 or DMA Function is in the <i>DL_Down</i> state 00_0010b = x2 00_1000b = x8 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> in DMA mode.	RsvdP	No	0
28	<b>Slot Clock Configuration</b> The DMA Function has the same value as the upstream Port. 0 = Indicates that the PEX 8619 uses an independent clock 1 = Indicates that the PEX 8619 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 18-24. 8Ch Device Capability 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Completion Timeout Range Supported</b>	RO	No	Fh
4	<b>Completion Timeout Disable Supported</b>	RO	No	1
31:5	<b>Reserved</b>	RsvdP	No	0-0h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-25. 90h Device Status and Control 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control 2</b>				
3:0	<b>Completion Timeout Value</b>	RW	Yes	0h
4	<b>Completion Timeout Disable</b>	RW	Yes	0
15:5	<b>Reserved</b>	RsvdP	No	0-0h
<b>Device Status 2</b>				
31:16	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-26. 98h Link Status and Control 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control 2</b>				
15:0	<b>Reserved</b>	RsvdP	No	0000h
<b>Link Status 2</b>				
16	<b>Current De-Emphasis Level</b> Reflects the de-emphasis level. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at 2.5 GT/s)	RO	No	0 (5.0 GT/s) 1 (2.5 GT/s)
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.9 DMA Interface Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – C4h)

This section details the Subsystem ID and Subsystem Vendor ID Capability registers in DMA mode. Table 18-7 defines the register map.

**Table 18-7. DMA Interface Subsystem ID and Subsystem Vendor ID Capability Register Map (DMA of Upstream 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>Reserved</i>								Next Capability Pointer (00h)				SSID/SSVID Capability ID (0Dh)							
Subsystem ID								Subsystem Vendor ID											
<i>Reserved</i>																			

**Register 18-27. A4h Subsystem Capability (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>SSID/SSVID Capability ID</b> Detects the 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 8619 DMA Interface Capabilities list The PEX 8619 DMA Interface Extended Capabilities list starts at offset 100h.	RO	Yes	00h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 18-28. A8h Subsystem ID and Subsystem Vendor ID (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Subsystem Vendor ID</b> The <i>Vendor ID</i> (offset 00h[15:0]) identifies the manufacturer of the PEX 8619, 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.	RO	Yes	10B5h
31:16	<b>Subsystem ID</b> The <i>Device ID</i> (offset 00h[31:16]) identifies the PEX 8619, and optionally the Subsystem ID in combination with the Subsystem Vendor ID, uniquely identifies the board or system. The <i>Subsystem ID</i> value 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 <i>Subsystem ID</i> value, in which case the Subsystem Vendor ID remains the PLX default value, 10B5h.  The <i>Subsystem Vendor ID</i> and <i>Subsystem ID</i> values are usually identical for all PEX 8619 Ports.	RO	Yes	8619h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.10 DMA Interface Base Address DMA Setup Register (Offsets C8h – FCh)

This section details the Base Address DMA Setup registers in DMA mode. Table 18-8 defines the register map.

**Table 18-8. DMA Interface Base Address DMA Setup Register Map (DMA of Upstream 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>Reserved</i>	C8h – E0h
	BAR0/1 DMA Setup	E4h
	<i>Reserved</i>	E8h – FCh

**Register 18-29. E4h BAR0/1 DMA Setup (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>BAR0/1 DMA Enable</b> 00b = Disables <b>BAR0 DMA</b> and <b>BAR1 DMA</b> 01b = <i>Reserved</i> 10b = Enables <b>BAR0 DMA</b> and disables <b>BAR1 DMA</b> <b>(BAR0 DMA</b> is a 32-bit BAR) 11b = Enables <b>BAR0 DMA</b> and <b>BAR1 DMA</b> <b>(BAR0/1 DMA</b> is a 64-bit BAR)	RW	No	10b
2	<b>BAR0 DMA Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RW	No	0
31:3	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 18.11 DMA Interface Device Serial Number Extended Capability Registers (Offsets 100h – 134h)

This section details the Device Serial Number Extended Capability registers in DMA mode. [Table 18-9](#) defines the register map.

**Table 18-9. DMA Interface Device Serial Number Extended Capability Register Map (DMA of Upstream 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		
Next Capability Offset (FB4h)	Capability Version (1h)	PCI Express Extended Capability ID (0003h)	100h
		Serial Number (Lower DW)	104h
		Serial Number (Upper DW)	108h
		<i>Reserved</i>	10Ch – 134h

**Register 18-30. 100h Device Serial Number Extended Capability Header (DMA of Upstream Port)**

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 r2.0</i> .	RO	Yes	0003h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r2.0</i> .	RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to FB4h, which is the <b>Advanced Error Reporting Extended Capability</b> structure.	RO	Yes	<a href="#">FB4h</a>

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-31. 104h Serial Number (Lower DW)  
(DMA of Upstream Port)**

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 Set by Serial EEPROM register initialization. Per the <i>PCI Express Base r2.0</i> , all switch Ports must contain the same value. The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64 <sup>TM</sup> ). The lower 24 bits are the Company ID value assigned by the IEEE registration authority, and the upper 40 bits are the Extension ID assigned by the identified Company.	RO	Yes	B5DF_0E00h

**Register 18-32. 108h Serial Number (Upper DW)  
(DMA of Upstream Port)**

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 Set by Serial EEPROM register initialization. Per the <i>PCI Express Base r2.0</i> , all switch Ports must contain the same value. The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64 <sup>TM</sup> ). The lower 24 bits are the Company ID value assigned by the IEEE registration authority, and the upper 40 bits are the Extension ID assigned by the identified Company.	RO	Yes	BA_8601_10h

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

## 18.12 Vendor-Specific Extended Capability Registers – DMA Channel (Offsets 1F4h – 5FCh)

This section details the Vendor-Specific Extended Capability DMA Channel registers in DMA mode. [Table 18-10](#) defines the register map.

**Table 18-10. DMA Interface Vendor-Specific Extended Capability DMA Channel Register Map (DMA of Upstream 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	
Next Capability Offset (000h)	Capability Version (1h)	PCI Express Extended Capability ID (000Bh)
		1F4h
	Vendor-Specific Header 1	1F8h
	DMA Global Control	1FCh
	DMA Channel 0 Source Address Lower	200h
	DMA Channel 0 Source Address Upper	204h
	DMA Channel 0 Destination Address Lower	208h
	DMA Channel 0 Destination Address Upper	20Ch
	DMA Channel 0 Transfer Size	210h
	DMA Channel 0 Descriptor Ring Address Lower	214h
	DMA Channel 0 Descriptor Ring Address Upper	218h
	DMA Channel 0 Next Descriptor Ring Address Lower	21Ch
	DMA Channel 0 Descriptor Ring Size	220h
	DMA Channel 0 Last Descriptor Address Lower	224h
	DMA Channel 0 Last Descriptor Transfer Size	228h
	DMA Channel 0 Bandwidth	22Ch
	<i>Factory Test Only</i>	230h
<i>Reserved</i>	DMA Channel 0 Maximum Prefetch Limit	234h
	DMA Channel 0 Control/Status	238h
	DMA Channel 0 Interrupt Control/Status	23Ch
	DMA Channel 0 Error Header Log 0	240h
	DMA Channel 0 Error Header Log 1	244h
	DMA Channel 0 Error Header Log 2	248h
	DMA Channel 0 Error Header Log 3	24Ch
	DMA Channel 0 Device-Specific Error Status	250h
	DMA Channel 0 Maximum Outstanding Read/Write Request	254h
<i>Reserved</i>		258h – 2FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 18-10. DMA Interface Vendor-Specific Extended Capability DMA Channel Register Map (DMA of Upstream Port) (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	
	DMA Channel 1 Source Address Lower	300h
	DMA Channel 1 Source Address Upper	304h
	DMA Channel 1 Destination Address Lower	308h
	DMA Channel 1 Destination Address Upper	30Ch
	DMA Channel 1 Transfer Size	310h
	DMA Channel 1 Descriptor Ring Address Lower	314h
	DMA Channel 1 Descriptor Ring Address Upper	318h
	DMA Channel 1 Next Descriptor Ring Address Lower	31Ch
	DMA Channel 1 Descriptor Ring Size	320h
	DMA Channel 1 Last Descriptor Address Lower	324h
	DMA Channel 1 Last Descriptor Transfer Size	328h
	DMA Channel 1 Bandwidth	32Ch
	<i>Factory Test Only</i>	330h
<i>Reserved</i>	DMA Channel 1 Maximum Prefetch Limit	334h
	DMA Channel 1 Control/Status	338h
	DMA Channel 1 Interrupt Control/Status	33Ch
	DMA Channel 1 Error Header Log 0	340h
	DMA Channel 1 Error Header Log 1	344h
	DMA Channel 1 Error Header Log 2	348h
	DMA Channel 1 Error Header Log 3	34Ch
	DMA Channel 1 Device-Specific Error Status	350h
	DMA Channel 1 Maximum Outstanding Read/Write Request	354h
<i>Reserved</i>		358h – 3FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 18-10. DMA Interface Vendor-Specific Extended Capability DMA Channel Register Map (DMA of Upstream Port) (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	
	DMA Channel 2 Source Address Lower	400h
	DMA Channel 2 Source Address Upper	404h
	DMA Channel 2 Destination Address Lower	408h
	DMA Channel 2 Destination Address Upper	40Ch
	DMA Channel 2 Transfer Size	410h
	DMA Channel 2 Descriptor Ring Address Lower	414h
	DMA Channel 2 Descriptor Ring Address Upper	418h
	DMA Channel 2 Next Descriptor Ring Address Lower	41Ch
	DMA Channel 2 Descriptor Ring Size	420h
	DMA Channel 2 Last Descriptor Address Lower	424h
	DMA Channel 2 Last Descriptor Transfer Size	428h
	DMA Channel 2 Bandwidth	42Ch
	<i>Factory Test Only</i>	430h
<i>Reserved</i>	DMA Channel 2 Maximum Prefetch Limit	434h
	DMA Channel 2 Control/Status	438h
	DMA Channel 2 Interrupt Control/Status	43Ch
	DMA Channel 2 Error Header Log 0	440h
	DMA Channel 2 Error Header Log 1	444h
	DMA Channel 2 Error Header Log 2	448h
	DMA Channel 2 Error Header Log 3	44Ch
	DMA Channel 2 Device-Specific Error Status	450h
	DMA Channel 2 Maximum Outstanding Read/Write Request	454h
<i>Reserved</i>		458h – 4FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 18-10. DMA Interface Vendor-Specific Extended Capability DMA Channel Register Map (DMA of Upstream Port) (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	
	DMA Channel 3 Source Address Lower	500h
	DMA Channel 3 Source Address Upper	504h
	DMA Channel 3 Destination Address Lower	508h
	DMA Channel 3 Destination Address Upper	50Ch
	DMA Channel 3 Transfer Size	510h
	DMA Channel 3 Descriptor Ring Address Lower	514h
	DMA Channel 3 Descriptor Ring Address Upper	518h
	DMA Channel 3 Next Descriptor Ring Address Lower	51Ch
	DMA Channel 3 Descriptor Ring Size	520h
	DMA Channel 3 Last Descriptor Address Lower	524h
	DMA Channel 3 Last Descriptor Transfer Size	528h
	DMA Channel 3 Bandwidth	52Ch
	<i>Factory Test Only</i>	530h
<i>Reserved</i>	DMA Channel 3 Maximum Prefetch Limit	534h
	DMA Channel 3 Control/Status	538h
	DMA Channel 3 Interrupt Control/Status	53Ch
	DMA Channel 3 Error Header Log 0	540h
	DMA Channel 3 Error Header Log 1	544h
	DMA Channel 3 Error Header Log 2	548h
	DMA Channel 3 Error Header Log 3	54Ch
	DMA Channel 3 Device-Specific Error Status	550h
	DMA Channel 3 Maximum Outstanding Read/Write Request	554h
<i>Reserved</i>		558h – 5FCh

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-33. 1F4h Vendor-Specific Extended Capability Header  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 000Bh, indicating that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	RO	Yes	000Bh
19:16	<b>Capability Version</b>	RO	Yes	1h
31:20	<b>Next Capability Offset</b> 000h = This extended capability is the last capability in the PEX 8619 Extended Capabilities list	RO	Yes	000h

**Register 18-34. 1F8h Vendor-Specific Header 1  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor-Specific ID</b> ID Number of this Extended Capability structure.	RO	Yes	0010h
19:16	<b>Vendor-Specific Rev</b> Version Number of this structure.	RO	Yes	0h
31:20	<b>Vendor-Specific Length</b> Quantity of bytes in the entire structure.	RO	Yes	410h

**Register 18-35. 1FCCh DMA Global Control  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>DMA Channel Mode</b> 00b = Four-Channel mode 01b = One-Channel mode 10b = Two-Channel mode 11b = <b>Reserved</b>	RW	Yes	00b
2	<b>DMA On-Chip Descriptor Mode</b> 0 = DMA Off-Chip Descriptor mode 1 = DMA On-Chip Descriptor mode <i>Note:</i> Refer to <a href="#">Section 10.6.3, “DMA On-Chip Descriptor Mode,”</a> and <a href="#">Section 10.6.3, “DMA On-Chip Descriptor Mode,”</a> for details regarding each mode.	RW	Yes	0
31:3	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-36. 200h DMA Channel 0 Source Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Source Address Lower</b> Lower 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-37. 204h DMA Channel 0 Source Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Source Address Upper</b> Upper 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-38. 208h DMA Channel 0 Destination Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b> Value is always the same as the lower two bits of the <b>DMA Channel 0 Source Address Lower</b> register (offset 200h[1:0]).	RsvdP	No	00b
31:2	<b>DMA Channel 0 Destination Address Lower</b> Lower 32 bits (including field [1:0] value) of Destination address where DMA Transfer Write starts.	RW	Yes	0-0h

**Register 18-39. 20Ch DMA Channel 0 Destination Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Destination Address Upper</b> Upper 32 bits of Destination address where DMA Transfer Write starts.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-40. 210h DMA Channel 0 Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 0 Transfer Size (Bytes)</b> Indicates the quantity of bytes to transfer during a DMA operation.	RW	Yes	0-0h
27	<i>Reserved</i>	RsvdP	No	0
28	<b>DMA Channel 0 Destination Address Constant</b>	RW	Yes	0
29	<b>DMA Channel 0 Source Address Constant</b>	RW	Yes	0
30	<b>DMA Channel 0 DMA Done Interrupt Enable</b>	RW	Yes	0
31	<b>DMA Channel 0 DMA Valid</b>	RW	Yes	0

**Register 18-41. 214h DMA Channel 0 Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<i>Reserved</i>	RsvdP	No	0h
31:6	<b>DMA Channel 0 Descriptor Ring Address Lower</b> Lower 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts. Ensures 64-bit alignment. <i>Note:</i> Bits [5:0] are always 00h.	RW	Yes	0-0h

**Register 18-42. 218h DMA Channel 0 Descriptor Ring Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Descriptor Ring Address Upper</b> Upper 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts.	RW	Yes	0000_0000h

**Register 18-43. 21Ch DMA Channel 0 Next Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<i>Reserved</i>	RsvdP	No	0h
31:4	<b>DMA Channel 0 Next Descriptor Ring Address Lower</b> Lower 32 bits (including field [3:0] value) of Descriptor Ring Base address of next Descriptor.	RW	Yes	0000_000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-44. 220h DMA Channel 0 Descriptor Ring Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Descriptor Ring Size</b> Indicates the quantity of Descriptors in the ring. The value includes both Normal Descriptor and Extended Descriptor formats. The ring must be greater than or equal to (Prefetch Limit Size + Depth of Descriptor FIFO + 1), to avoid the possibility of the Descriptor prefetching looparound faster than the Completion Status writeback for that Descriptor in Ring Delay mode.	RW	Yes	0000_0000h

**Register 18-45. 224h DMA Channel 0 Last Descriptor Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 0 Last Descriptor Address Lower</b> Lower 32 bits (including field [3:0] value) of the last Descriptor address. Software polls this register to determine the Descriptor progress for the current DMA transfer.	RO	Yes	0000_000h

**Register 18-46. 228h DMA Channel 0 Last Descriptor Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 0 Last Descriptor Transfer Size</b> Last Descriptor's Transfer size, which corresponds to the last Descriptor Address register.	RO	Yes	0-0h
31:27	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-47. 22Ch DMA Channel 0 Bandwidth  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>DMA Channel 0 DMA Clocks to Transfer</b> Defines the wait period, in 4-ns clock increments, for intervals between two Memory Read Requests issued by the DMA Controller.	RW	Yes	0000h
16	<b>DMA Channel 0 DMA Immediate Pause</b>	RW	Yes	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-48. 234h DMA Channel 0 Maximum Prefetch Limit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<p><b>DMA Channel 0 Maximum Prefetch Limit</b>            Indicates the maximum quantity of Descriptors that can be prefetched for this channel.            A value of 0 = 256 in One-Channel mode, when this limit is reached, DMA Descriptor prefetching stops, and resumes only after the value drops below the limit, to an integer multiple of 4 value.            For a value of 1, only one Descriptor is prefetched. When that one Descriptor is active, a second Descriptor is prefetched.            For a value of 4, four Descriptors are prefetched. When the fourth Descriptor is active, another four Descriptors are prefetched.            For values of 8, 12, 16, and so forth (up to FFh), when the count is 4 less than the Maximum Prefetch Limit, four more Descriptors are prefetched.  <b>Note:</b> <i>The value can be programmed only with a multiple of 4 or 1.</i></p>	RW	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-49. 238h DMA Channel 0 Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 0 DMA Graceful Pause</b>	RW	Yes	0
1	<b>DMA Channel 0 DMA Abort</b>	RW	Yes	0
2	<b>DMA Channel 0 DMA Completion Status Write Back Enable</b>	RW	Yes	0
3	<b>DMA Channel 0 DMA Start</b>	RW	Yes	0
4	<b>DMA Channel 0 DMA Descriptor Mode Select</b> 0 = DMA Block mode 1 = DMA Ring Descriptor mode	RW	Yes	0
5	<b>DMA Channel 0 DMA Ring Stop Mode</b> 0 = Descriptor polling continues after the Descriptor's current round finishes 1 = Descriptor Ring stops after the Descriptor's current round finishes	RW	Yes	0
6	<b>DMA Channel 0 DMA Channel Reset</b> <i>Not supported</i> Always read as 0.	RW	Yes	0
7	<b>Reserved</b>	RsvdP	No	0
8	<b>DMA Channel 0 DMA Descriptor Invalid Status</b>	RW1C	Yes	0
9	<b>DMA Channel 0 DMA Graceful Pause Done Status</b>	RW1C	Yes	0
10	<b>DMA Channel 0 DMA Abort Done Status</b>	RW1C	Yes	0
11	<b>Factory Test Only</b>	RW1C	Yes	0
12	<b>DMA Channel 0 DMA Immediate Pause Done Status</b>	RW1C	Yes	0
15:13	<b>DMA Channel 0 DMA Descriptor Delay Period for Ring Wraparound</b> DMA Descriptor polling period. 000b = Continuously, without any wait (default) 001b = 1 µs 010b = 2 µs 011b = 8 µs 100b = 32 µs 101b = 128 µs 110b = 512 µs 111b = 1 ms	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-49. 238h DMA Channel 0 Control/Status  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>DMA Channel 0 DMA Source Maximum Transfer Size</b> 000b = 64 bytes 001b = 128 bytes 010b = 256 bytes 011b = 512 bytes 100b = 1 KB 101b = 2 KB 111b = 4 bytes  All other encodings are <i>reserved</i> .  <i>Note:</i> The value of this field must not be greater than the Maximum Payload Size indicated by the <a href="#">Device Control register Maximum Payload Size field (DMA of Upstream Port, offset 70h[7:5])</a> .	RW	Yes	000b
21:19	<b>DMA Channel 0 DMA Traffic Class</b>	RW	Yes	000b
22	<b>DMA Channel 0 DMA Relaxed Ordering for Descriptor Read Request</b>	RW	Yes	0
23	<b>DMA Channel 0 DMA Relaxed Ordering for Data Read Request</b>	RW	Yes	0
24	<b>DMA Channel 0 DMA Relaxed Ordering for Descriptor Write</b>	RW	Yes	0
25	<b>DMA Channel 0 DMA Relaxed Ordering for Data Write</b>	RW	Yes	0
26	<b>DMA Channel 0 DMA No Snoop for Descriptor Read Request</b>	RW	Yes	0
27	<b>DMA Channel 0 DMA No Snoop for Data Read Request</b>	RW	Yes	0
28	<b>DMA Channel 0 DMA No Snoop for Descriptor Write</b>	RW	Yes	0
29	<b>DMA Channel 0 DMA No Snoop for Data Write</b>	RW	Yes	0
30	<b>DMA Channel 0 In-Progress</b>	RO	No	0
31	<b>DMA Channel 0 DMA Header Logging Valid</b>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-50. 23Ch DMA Channel 0 Interrupt Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 0 DMA Error Interrupt Enable</b>	RW	Yes	0
1	<b>DMA Channel 0 DMA Invalid Descriptor Interrupt Enable</b>	RW	Yes	0
2	<b>Reserved</b>	RsvdP	No	0
3	<b>DMA Channel 0 DMA Abort Done Interrupt Enable</b>	RW	Yes	0
4	<b>DMA Channel 0 DMA Graceful Pause Done Interrupt Enable</b>	RW	Yes	0
5	<b>DMA Channel 0 DMA Immediate Pause Done Interrupt Enable</b>	RW	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>DMA Channel 0 DMA Error Interrupt Status</b>	RW1C	Yes	0
17	<b>DMA Channel 0 DMA Invalid Descriptor Interrupt Status</b>	RW1C	Yes	0
18	<b>DMA Channel 0 DMA Descriptor Done Interrupt Status</b>	RW1C	Yes	0
19	<b>DMA Channel 0 DMA Abort Done Interrupt Status</b>	RW1C	Yes	0
20	<b>DMA Channel 0 DMA Graceful Pause Done Interrupt Status</b>	RW1C	Yes	0
21	<b>DMA Channel 0 DMA Immediate Pause Done Interrupt Status</b>	RW1C	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-51. 240h DMA Channel 0 Error Header Log 0  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Error Header Log 0</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-52. 244h DMA Channel 0 Error Header Log 1  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Error Header Log 1</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-53. 248h DMA Channel 0 Error Header Log 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Error Header Log 2</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-54. 24Ch DMA Channel 0 Error Header Log 3  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 0 Error Header Log 3</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-55. 250h DMA Channel 0 Device-Specific Error Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 0 Data Read Error Status</b>	RW1CS	Yes	0
1	<b>DMA Channel 0 Data Write Error Status</b>	RW1CS	Yes	0
2	<b>DMA Channel 0 Descriptor Read Error Status</b>	RW1CS	Yes	0
3	<b>DMA Channel 0 Descriptor Write Error Status</b>	RW1CS	Yes	0
31:4	<i>Reserved</i>	RsvdP	No	0000_000h

**Register 18-56. 254h DMA Channel 0 Maximum Outstanding Read/Write Request  
(DMA, Upstream Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b>DMA Channel 0 Maximum Outstanding Request Supported for DMA Data Read</b>	RW	Yes	3Fh
15:6	<i>Reserved</i>	RsvdP	No	0-0h
21:16	<b>DMA Channel 0 Maximum Outstanding Request Supported for DMA Data Write</b> This register is invalid, and should always be programmed to 00h.	RW	Yes	00h
31:22	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-57. 300h DMA Channel 1 Source Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Source Address Lower</b> Lower 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-58. 304h DMA Channel 1 Source Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Source Address Upper</b> Upper 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-59. 308h DMA Channel 1 Destination Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b> Value is always the same as the lower two bits of the <b>DMA Channel 1 Source Address Lower</b> register (offset 300h[1:0]).	RsvdP	No	00b
31:2	<b>DMA Channel 1 Destination Address Lower</b> Lower 32 bits (including field [1:0] value) of Destination address where DMA Transfer Write starts.	RW	Yes	0-0h

**Register 18-60. 30Ch DMA Channel 1 Destination Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Destination Address Upper</b> Upper 32 bits of Destination address where DMA Transfer Write starts.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-61. 310h DMA Channel 1 Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 1 Transfer Size (Bytes)</b> Indicates the quantity of bytes to transfer during a DMA operation.	RW	Yes	0-0h
27	<b>Reserved</b>	RsvdP	No	0
28	<b>DMA Channel 1 Destination Address Constant</b>	RW	Yes	0
29	<b>DMA Channel 1 Source Address Constant</b>	RW	Yes	0
30	<b>DMA Channel 1 DMA Done Interrupt Enable</b>	RW	Yes	0
31	<b>DMA Channel 1 DMA Valid</b>	RW	Yes	0

**Register 18-62. 314h DMA Channel 1 Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b>Reserved</b>	RsvdP	No	00h
31:6	<b>DMA Channel 1 Descriptor Ring Address Lower</b> Lower 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts. Ensures 64-bit alignment. <i>Note:</i> Bits [5:0] are always 00h.	RW	Yes	0-0h

**Register 18-63. 318h DMA Channel 1 Descriptor Ring Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Descriptor Ring Address Upper</b> Upper 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts.	RW	Yes	0000_0000h

**Register 18-64. 31Ch DMA Channel 1 Next Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 1 Next Descriptor Ring Address Lower</b> Lower 32 bits (including field [3:0] value) of Descriptor Ring Base address of next Descriptor.	RW	Yes	0000_000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-65. 320h DMA Channel 1 Descriptor Ring Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Descriptor Ring Size</b> Indicates the quantity of Descriptors in the ring. The value includes both Normal Descriptor and Extended Descriptor formats. The ring must be greater than or equal to (Prefetch Limit Size + Depth of Descriptor FIFO + 1), to avoid the possibility of the Descriptor prefetching looparound faster than the Completion Status writeback for that Descriptor in Ring Delay mode.	RW	Yes	0000_0000h

**Register 18-66. 324h DMA Channel 1 Last Descriptor Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 1 Last Descriptor Address Lower</b> Lower 32 bits (including field [3:0] value) of the last Descriptor address. Software polls this register to determine the Descriptor progress for the current DMA transfer.	RO	Yes	0000_000h

**Register 18-67. 328h DMA Channel 1 Last Descriptor Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 1 Last Descriptor Transfer Size</b> Last Descriptor's Transfer size, which corresponds to the last Descriptor Address register.	RO	Yes	0-0h
31:27	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-68. 32Ch DMA Channel 1 Bandwidth  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>DMA Channel 1 DMA Clocks to Transfer</b> Defines the wait period, in 4-ns clock increments, for intervals between two Memory Read Requests issued by the DMA Controller.	RW	Yes	0000h
16	<b>DMA Channel 1 DMA Immediate Pause</b>	RW	Yes	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-69. 334h DMA Channel 1 Maximum Prefetch Limit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<p><b>DMA Channel 1 Maximum Prefetch Limit</b>            Indicates the maximum quantity of Descriptors that can be prefetched for this channel.            A value of 0 = 256 in One-Channel mode, when this limit is reached, DMA Descriptor prefetching stops, and resumes only after the value drops below the limit, to an integer multiple of 4 value.            For a value of 1, only one Descriptor is prefetched. When that one Descriptor is active, a second Descriptor is prefetched.            For a value of 4, four Descriptors are prefetched. When the fourth Descriptor is active, another four Descriptors are prefetched.            For values of 8, 12, 16, and so forth (up to FFh), when the count is 4 less than the Maximum Prefetch Limit, four more Descriptors are prefetched.</p> <p><i>Note: The value can be programmed only with a multiple of 4 or 1.</i></p>	RW	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-70. 338h DMA Channel 1 Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 1 DMA Graceful Pause</b>	RW	Yes	0
1	<b>DMA Channel 1 DMA Abort</b>	RW	Yes	0
2	<b>DMA Channel 1 DMA Completion Status Write Back Enable</b>	RW	Yes	0
3	<b>DMA Channel 1 DMA Start</b>	RW	Yes	0
4	<b>DMA Channel 1 DMA Descriptor Mode Select</b> 0 = DMA Block mode 1 = DMA Ring Descriptor mode	RW	Yes	0
5	<b>DMA Channel 1 DMA Ring Stop Mode</b> 0 = Descriptor polling continues after the Descriptor's current round finishes 1 = Descriptor Ring stops after the Descriptor's current round finishes	RW	Yes	0
6	<b>DMA Channel 1 DMA Channel Reset</b> <i>Not supported</i> Always read as 0.	RW	Yes	0
7	<b>Reserved</b>	RsvdP	No	0
8	<b>DMA Channel 1 DMA Descriptor Invalid Status</b>	RW1C	Yes	0
9	<b>DMA Channel 1 DMA Graceful Pause Done Status</b>	RW1C	Yes	0
10	<b>DMA Channel 1 DMA Abort Done Status</b>	RW1C	Yes	0
11	<b>Factory Test Only</b>	RW1C	Yes	0
12	<b>DMA Channel 1 DMA Immediate Pause Done Status</b>	RW1C	Yes	0
15:13	<b>DMA Channel 1 DMA Descriptor Delay Period for Ring Wraparound</b> DMA Descriptor polling period. 000b = Continuously, without any wait (default) 001b = 1 µs 010b = 2 µs 011b = 8 µs 100b = 32 µs 101b = 128 µs 110b = 512 µs 111b = 1 ms	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-70. 338h DMA Channel 1 Control/Status  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>DMA Channel 1 DMA Source Maximum Transfer Size</b> 000b = 64 bytes 001b = 128 bytes 010b = 256 bytes 011b = 512 bytes 100b = 1 KB 101b = 2 KB 111b = 4 bytes  All other encodings are <i>reserved</i> .  <i>Note:</i> The value of this field must not be greater than the Maximum Payload Size indicated by the <a href="#">Device Control register Maximum Payload Size field (DMA of Upstream Port, offset 70h[7:5])</a> .	RW	Yes	000b
21:19	<b>DMA Channel 1 DMA Traffic Class</b>	RW	Yes	000b
22	<b>DMA Channel 1 DMA Relaxed Ordering for Descriptor Read Request</b>	RW	Yes	0
23	<b>DMA Channel 1 DMA Relaxed Ordering for Data Read Request</b>	RW	Yes	0
24	<b>DMA Channel 1 DMA Relaxed Ordering for Descriptor Write</b>	RW	Yes	0
25	<b>DMA Channel 1 DMA Relaxed Ordering for Data Write</b>	RW	Yes	0
26	<b>DMA Channel 1 DMA No Snoop for Descriptor Read Request</b>	RW	Yes	0
27	<b>DMA Channel 1 DMA No Snoop for Data Read Request</b>	RW	Yes	0
28	<b>DMA Channel 1 DMA No Snoop for Descriptor Write</b>	RW	Yes	0
29	<b>DMA Channel 1 DMA No Snoop for Data Write</b>	RW	Yes	0
30	<b>DMA Channel 1 In-Progress</b>	RO	No	0
31	<b>DMA Channel 1 DMA Header Logging Valid</b>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-71. 33Ch DMA Channel 1 Interrupt Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 1 DMA Error Interrupt Enable</b>	RW	Yes	0
1	<b>DMA Channel 1 DMA Invalid Descriptor Interrupt Enable</b>	RW	Yes	0
2	<b>Reserved</b>	RsvdP	No	0
3	<b>DMA Channel 1 DMA Abort Done Interrupt Enable</b>	RW	Yes	0
4	<b>DMA Channel 1 DMA Graceful Pause Done Interrupt Enable</b>	RW	Yes	0
5	<b>DMA Channel 1 DMA Immediate Pause Done Interrupt Enable</b>	RW	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>DMA Channel 1 DMA Error Interrupt Status</b>	RW1C	Yes	0
17	<b>DMA Channel 1 DMA Invalid Descriptor Interrupt Status</b>	RW1C	Yes	0
18	<b>DMA Channel 1 DMA Descriptor Done Interrupt Status</b>	RW1C	Yes	0
19	<b>DMA Channel 1 DMA Abort Done Interrupt Status</b>	RW1C	Yes	0
20	<b>DMA Channel 1 DMA Graceful Pause Done Interrupt Status</b>	RW1C	Yes	0
21	<b>DMA Channel 1 DMA Immediate Pause Done Interrupt Status</b>	RW1C	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-72. 340h DMA Channel 1 Error Header Log 0  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Error Header Log 0</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-73. 344h DMA Channel 1 Error Header Log 1  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Error Header Log 1</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-74. 348h DMA Channel 1 Error Header Log 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Error Header Log 2</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-75. 34Ch DMA Channel 1 Error Header Log 3  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 1 Error Header Log 3</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-76. 350h DMA Channel 1 Device-Specific Error Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	DMA Channel 1 Data Read Error Status	RW1CS	Yes	0
1	DMA Channel 1 Data Write Error Status	RW1CS	Yes	0
2	DMA Channel 1 Descriptor Read Error Status	RW1CS	Yes	0
3	DMA Channel 1 Descriptor Write Error Status	RW1CS	Yes	0
31:4	<i>Reserved</i>	RsvdP	No	0000_000h

**Register 18-77. 354h DMA Channel 1 Maximum Outstanding Read/Write Request  
(DMA, Upstream Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	DMA Channel 1 Maximum Outstanding Request Supported for DMA Data Read	RW	Yes	3Fh
15:6	<i>Reserved</i>	RsvdP	No	0-0h
21:16	DMA Channel 1 Maximum Outstanding Request Supported for DMA Data Write This register is invalid, and should always be programmed to 00h.	RW	Yes	00h
31:22	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-78. 400h DMA Channel 2 Source Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Source Address Lower</b> Lower 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-79. 404h DMA Channel 2 Source Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Source Address Upper</b> Upper 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-80. 408h DMA Channel 2 Destination Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b> Value is always the same as the lower two bits of the <b>DMA Channel 2 Source Address Lower</b> register (offset 400h[1:0]).	RsvdP	No	00b
31:2	<b>DMA Channel 2 Destination Address Lower</b> Lower 32 bits (including field [1:0] value) of Destination address where DMA Transfer Write starts.	RW	Yes	0-0h

**Register 18-81. 40Ch DMA Channel 2 Destination Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Destination Address Upper</b> Upper 32 bits of Destination address where DMA Transfer Write starts.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-82. 410h DMA Channel 2 Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 2 Transfer Size (Bytes)</b> Indicates the quantity of bytes to transfer during a DMA operation.	RW	Yes	0-0h
27	<i>Reserved</i>	RsvdP	No	0
28	<b>DMA Channel 2 Destination Address Constant</b>	RW	Yes	0
29	<b>DMA Channel 2 Source Address Constant</b>	RW	Yes	0
30	<b>DMA Channel 2 DMA Done Interrupt Enable</b>	RW	Yes	0
31	<b>DMA Channel 2 DMA Valid</b>	RW	Yes	0

**Register 18-83. 414h DMA Channel 2 Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<i>Reserved</i>	RsvdP	No	00h
31:6	<b>DMA Channel 2 Descriptor Ring Address Lower</b> Lower 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts. Ensures 64-bit alignment. <i>Note:</i> Bits [5:0] are always 00h.	RW	Yes	0-0h

**Register 18-84. 418h DMA Channel 2 Descriptor Ring Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Descriptor Ring Address Upper</b> Upper 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts.	RW	Yes	0000_0000h

**Register 18-85. 41Ch DMA Channel 2 Next Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<i>Reserved</i>	RsvdP	No	0h
31:4	<b>DMA Channel 2 Next Descriptor Ring Address Lower</b> Lower 32 bits (including field [3:0] value) of Descriptor Ring Base address of next Descriptor.	RW	Yes	0000_000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-86. 420h DMA Channel 2 Descriptor Ring Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Descriptor Ring Size</b> Indicates the quantity of Descriptors in the ring. The value includes both Normal Descriptor and Extended Descriptor formats. The ring must be greater than or equal to (Prefetch Limit Size + Depth of Descriptor FIFO + 1), to avoid the possibility of the Descriptor prefetching looparound faster than the Completion Status writeback for that Descriptor in Ring Delay mode.	RW	Yes	0000_0000h

**Register 18-87. 424h DMA Channel 2 Last Descriptor Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 2 Last Descriptor Address Lower</b> Lower 32 bits (including field [3:0] value) of the last Descriptor address. Software polls this register to determine the Descriptor progress for the current DMA transfer.	RO	Yes	0000_000h

**Register 18-88. 428h DMA Channel 2 Last Descriptor Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 2 Last Descriptor Transfer Size</b> Last Descriptor's Transfer size, which corresponds to the last Descriptor Address register.	RO	Yes	0-0h
31:27	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-89. 42Ch DMA Channel 2 Bandwidth  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>DMA Channel 2 DMA Clocks to Transfer</b> Defines the wait period, in 4-ns clock increments, for intervals between two Memory Read Requests issued by the DMA Controller.	RW	Yes	0000h
16	<b>DMA Channel 2 DMA Immediate Pause</b>	RW	Yes	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-90. 434h DMA Channel 2 Maximum Prefetch Limit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<p><b>DMA Channel 2 Maximum Prefetch Limit</b>            Indicates the maximum quantity of Descriptors that can be prefetched for this channel.            A value of 0 = 256 in One-Channel mode, when this limit is reached, DMA Descriptor prefetching stops, and resumes only after the value drops below the limit, to an integer multiple of 4 value.            For a value of 1, only one Descriptor is prefetched. When that one Descriptor is active, a second Descriptor is prefetched.            For a value of 4, four Descriptors are prefetched. When the fourth Descriptor is active, another four Descriptors are prefetched.            For values of 8, 12, 16, and so forth (up to FFh), when the count is 4 less than the Maximum Prefetch Limit, four more Descriptors are prefetched.  <b>Note:</b> <i>The value can be programmed only with a multiple of 4 or 1.</i></p>	RW	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-91. 438h DMA Channel 2 Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 2 DMA Graceful Pause</b>	RW	Yes	0
1	<b>DMA Channel 2 DMA Abort</b>	RW	Yes	0
2	<b>DMA Channel 2 DMA Completion Status Write Back Enable</b>	RW	Yes	0
3	<b>DMA Channel 2 DMA Start</b>	RW	Yes	0
4	<b>DMA Channel 2 DMA Descriptor Mode Select</b> 0 = DMA Block mode 1 = DMA Ring Descriptor mode	RW	Yes	0
5	<b>DMA Channel 2 DMA Ring Stop Mode</b> 0 = Descriptor polling continues after the Descriptor's current round finishes 1 = Descriptor Ring stops after the Descriptor's current round finishes	RW	Yes	0
6	<b>DMA Channel 2 DMA Channel Reset</b> <i>Not supported</i> Always read as 0.	RW	Yes	0
7	<b>Reserved</b>	RsvdP	No	0
8	<b>DMA Channel 2 DMA Descriptor Invalid Status</b>	RW1C	Yes	0
9	<b>DMA Channel 2 DMA Graceful Pause Done Status</b>	RW1C	Yes	0
10	<b>DMA Channel 2 DMA Abort Done Status</b>	RW1C	Yes	0
11	<b>Factory Test Only</b>	RW1C	Yes	0
12	<b>DMA Channel 2 DMA Immediate Pause Done Status</b>	RW1C	Yes	0
15:13	<b>DMA Channel 2 DMA Descriptor Delay Period for Ring Wraparound</b> DMA Descriptor polling period. 000b = Continuously, without any wait (default) 001b = 1 µs 010b = 2 µs 011b = 8 µs 100b = 32 µs 101b = 128 µs 110b = 512 µs 111b = 1 ms	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-91. 438h DMA Channel 2 Control/Status  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>DMA Channel 2 DMA Source Maximum Transfer Size</b> 000b = 64 bytes 001b = 128 bytes 010b = 256 bytes 011b = 512 bytes 100b = 1 KB 101b = 2 KB 111b = 4 bytes  All other encodings are <i>reserved</i> .  <i>Note:</i> The value of this field must not be greater than the Maximum Payload Size indicated by the <a href="#">Device Control register Maximum Payload Size field (DMA of Upstream Port, offset 70h[7:5])</a> .	RW	Yes	000b
21:19	<b>DMA Channel 2 DMA Traffic Class</b>	RW	Yes	000b
22	<b>DMA Channel 2 DMA Relaxed Ordering for Descriptor Read Request</b>	RW	Yes	0
23	<b>DMA Channel 2 DMA Relaxed Ordering for Data Read Request</b>	RW	Yes	0
24	<b>DMA Channel 2 DMA Relaxed Ordering for Descriptor Write</b>	RW	Yes	0
25	<b>DMA Channel 2 DMA Relaxed Ordering for Data Write</b>	RW	Yes	0
26	<b>DMA Channel 2 DMA No Snoop for Descriptor Read Request</b>	RW	Yes	0
27	<b>DMA Channel 2 DMA No Snoop for Data Read Request</b>	RW	Yes	0
28	<b>DMA Channel 2 DMA No Snoop for Descriptor Write</b>	RW	Yes	0
29	<b>DMA Channel 2 DMA No Snoop for Data Write</b>	RW	Yes	0
30	<b>DMA Channel 2 In-Progress</b>	RO	No	0
31	<b>DMA Channel 2 DMA Header Logging Valid</b>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-92. 43Ch DMA Channel 2 Interrupt Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 2 DMA Error Interrupt Enable</b>	RW	Yes	0
1	<b>DMA Channel 2 DMA Invalid Descriptor Interrupt Enable</b>	RW	Yes	0
2	<b>Reserved</b>	RsvdP	No	0
3	<b>DMA Channel 2 DMA Abort Done Interrupt Enable</b>	RW	Yes	0
4	<b>DMA Channel 2 DMA Graceful Pause Done Interrupt Enable</b>	RW	Yes	0
5	<b>DMA Channel 2 DMA Immediate Pause Done Interrupt Enable</b>	RW	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>DMA Channel 2 DMA Error Interrupt Status</b>	RW1C	Yes	0
17	<b>DMA Channel 2 DMA Invalid Descriptor Interrupt Status</b>	RW1C	Yes	0
18	<b>DMA Channel 2 DMA Descriptor Done Interrupt Status</b>	RW1C	Yes	0
19	<b>DMA Channel 2 DMA Abort Done Interrupt Status</b>	RW1C	Yes	0
20	<b>DMA Channel 2 DMA Graceful Pause Done Interrupt Status</b>	RW1C	Yes	0
21	<b>DMA Channel 2 DMA Immediate Pause Done Interrupt Status</b>	RW1C	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-93. 440h DMA Channel 2 Error Header Log 0  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Error Header Log 0</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-94. 444h DMA Channel 2 Error Header Log 1  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Error Header Log 1</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-95. 448h DMA Channel 2 Error Header Log 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Error Header Log 2</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-96. 44Ch DMA Channel 2 Error Header Log 3  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 2 Error Header Log 3</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-97. 450h DMA Channel 2 Device-Specific Error Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 2 Data Read Error Status</b>	RW1CS	Yes	0
1	<b>DMA Channel 2 Data Write Error Status</b>	RW1CS	Yes	0
2	<b>DMA Channel 2 Descriptor Read Error Status</b>	RW1CS	Yes	0
3	<b>DMA Channel 2 Descriptor Write Error Status</b>	RW1CS	Yes	0
31:4	<i>Reserved</i>	RsvdP	No	0000_000h

**Register 18-98. 454h DMA Channel 2 Maximum Outstanding Read/Write Request  
(DMA, Upstream Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b>DMA Channel 2 Maximum Outstanding Request Supported for DMA Data Read</b>	RW	Yes	3Fh
15:6	<i>Reserved</i>	RsvdP	No	0-0h
21:16	<b>DMA Channel 2 Maximum Outstanding Request Supported for DMA Data Write</b> This register is invalid, and should always be programmed to 00h.	RW	Yes	00h
31:22	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-99. 500h DMA Channel 3 Source Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Source Address Lower</b> Lower 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-100. 504h DMA Channel 3 Source Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Source Address Upper</b> Upper 32 bits of Source address where DMA Transfer Read starts.	RW	Yes	0000_0000h

**Register 18-101. 508h DMA Channel 3 Destination Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Reserved</b> Value is always the same as the lower two bits of the <b>DMA Channel 3 Source Address Lower</b> register (offset 500h[1:0]).	RsvdP	No	00b
31:2	<b>DMA Channel 3 Destination Address Lower</b> Lower 32 bits (including field [1:0] value) of Destination address where DMA Transfer Write starts.	RW	Yes	0-0h

**Register 18-102. 50Ch DMA Channel 3 Destination Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Destination Address Upper</b> Upper 32 bits of Destination address where DMA Transfer Write starts.	RW	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-103. 510h DMA Channel 3 Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 3 Transfer Size (Bytes)</b> Indicates the quantity of bytes to transfer during a DMA operation.	RW	Yes	0-0h
27	<b>Reserved</b>	RsvdP	No	0
28	<b>DMA Channel 3 Destination Address Constant</b>	RW	Yes	0
29	<b>DMA Channel 3 Source Address Constant</b>	RW	Yes	0
30	<b>DMA Channel 3 DMA Done Interrupt Enable</b>	RW	Yes	0
31	<b>DMA Channel 3 DMA Valid</b>	RW	Yes	0

**Register 18-104. 514h DMA Channel 3 Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b>Reserved</b>	RsvdP	No	00h
31:6	<b>DMA Channel 3 Descriptor Ring Address Lower</b> Lower 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts. Ensures 64-bit alignment. <i>Note:</i> Bits [5:0] are always 00h.	RW	Yes	0-0h

**Register 18-105. 518h DMA Channel 3 Descriptor Ring Address Upper  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Descriptor Ring Address Upper</b> Upper 32 bits of Descriptor Ring Base address where DMA Descriptor Read starts.	RW	Yes	0000_0000h

**Register 18-106. 51Ch DMA Channel 3 Next Descriptor Ring Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 3 Next Descriptor Ring Address Lower</b> Lower 32 bits (including field [3:0] value) of Descriptor Ring Base address of next Descriptor.	RW	Yes	0000_000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-107. 520h DMA Channel 3 Descriptor Ring Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Descriptor Ring Size</b> Indicates the quantity of Descriptors in the ring. The value includes both Normal Descriptor and Extended Descriptor formats. The ring must be greater than or equal to (Prefetch Limit Size + Depth of Descriptor FIFO + 1), to avoid the possibility of the Descriptor prefetching looparound faster than the Completion Status writeback for that Descriptor in Ring Delay mode.	RW	Yes	0000_0000h

**Register 18-108. 524h DMA Channel 3 Last Descriptor Address Lower  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
31:4	<b>DMA Channel 3 Last Descriptor Address Lower</b> Lower 32 bits (including field [3:0] value) of the last Descriptor address. Software polls this register to determine the Descriptor progress for the current DMA transfer.	RO	Yes	0000_000h

**Register 18-109. 528h DMA Channel 3 Last Descriptor Transfer Size  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:0	<b>DMA Channel 3 Last Descriptor Transfer Size</b> Last Descriptor's Transfer size, which corresponds to the last Descriptor Address register.	RO	Yes	0-0h
31:27	<b>Reserved</b>	RsvdP	No	0-0h

**Register 18-110. 52Ch DMA Channel 3 Bandwidth  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>DMA Channel 3 DMA Clocks to Transfer</b> Defines the wait period, in 4-ns clock increments, for intervals between two Memory Read Requests issued by the DMA Controller.	RW	Yes	0000h
16	<b>DMA Channel 3 DMA Immediate Pause</b>	RW	Yes	0
31:17	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-111. 534h DMA Channel 3 Maximum Prefetch Limit  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<p><b>DMA Channel 3 Maximum Prefetch Limit</b>            Indicates the maximum quantity of Descriptors that can be prefetched for this channel.            A value of 0 = 256 in One-Channel mode, when this limit is reached, DMA Descriptor prefetching stops, and resumes only after the value drops below the limit, to an integer multiple of 4 value.            For a value of 1, only one Descriptor is prefetched. When that one Descriptor is active, a second Descriptor is prefetched.            For a value of 4, four Descriptors are prefetched. When the fourth Descriptor is active, another four Descriptors are prefetched.            For values of 8, 12, 16, and so forth (up to FFh), when the count is 4 less than the Maximum Prefetch Limit, four more Descriptors are prefetched.</p> <p><i>Note: The value can be programmed only with a multiple of 4 or 1.</i></p>	RW	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-112. 538h DMA Channel 3 Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 3 DMA Graceful Pause</b>	RW	Yes	0
1	<b>DMA Channel 3 DMA Abort</b>	RW	Yes	0
2	<b>DMA Channel 3 DMA Completion Status Write Back Enable</b>	RW	Yes	0
3	<b>DMA Channel 3 DMA Start</b>	RW	Yes	0
4	<b>DMA Channel 3 DMA Descriptor Mode Select</b> 0 = DMA Block mode 1 = DMA Ring Descriptor mode	RW	Yes	0
5	<b>DMA Channel 3 DMA Ring Stop Mode</b> 0 = Descriptor polling continues after the Descriptor's current round finishes 1 = Descriptor Ring stops after the Descriptor's current round finishes	RW	Yes	0
6	<b>DMA Channel 3 DMA Channel Reset</b> <i>Not supported</i> Always read as 0.	RW	Yes	0
7	<b>Reserved</b>	RsvdP	No	0
8	<b>DMA Channel 3 DMA Descriptor Invalid Status</b>	RW1C	Yes	0
9	<b>DMA Channel 3 DMA Graceful Pause Done Status</b>	RW1C	Yes	0
10	<b>DMA Channel 3 DMA Abort Done Status</b>	RW1C	Yes	0
11	<b>Factory Test Only</b>	RW1C	Yes	0
12	<b>DMA Channel 3 DMA Immediate Pause Done Status</b>	RW1C	Yes	0
15:13	<b>DMA Channel 3 DMA Descriptor Delay Period for Ring Wraparound</b> DMA Descriptor polling period. 000b = Continuously, without any wait (default) 001b = 1 µs 010b = 2 µs 011b = 8 µs 100b = 32 µs 101b = 128 µs 110b = 512 µs 111b = 1 ms	RW	Yes	000b

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Register 18-112. 538h DMA Channel 3 Control/Status  
(DMA of Upstream Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
18:16	<b>DMA Channel 3 DMA Source Maximum Transfer Size</b> 000b = 64 bytes 001b = 128 bytes 010b = 256 bytes 011b = 512 bytes 100b = 1 KB 101b = 2 KB 111b = 4 bytes  All other encodings are <i>reserved</i> .  <i>Note:</i> The value of this field must not be greater than the Maximum Payload Size indicated by the <a href="#">Device Control register Maximum Payload Size field (DMA of Upstream Port, offset 70h[7:5])</a> .	RW	Yes	000b
21:19	<b>DMA Channel 3 DMA Traffic Class</b>	RW	Yes	000b
22	<b>DMA Channel 3 DMA Relaxed Ordering for Descriptor Read Request</b>	RW	Yes	0
23	<b>DMA Channel 3 DMA Relaxed Ordering for Data Read Request</b>	RW	Yes	0
24	<b>DMA Channel 3 DMA Relaxed Ordering for Descriptor Write</b>	RW	Yes	0
25	<b>DMA Channel 3 DMA Relaxed Ordering for Data Write</b>	RW	Yes	0
26	<b>DMA Channel 3 DMA No Snoop for Descriptor Read Request</b>	RW	Yes	0
27	<b>DMA Channel 3 DMA No Snoop for Data Read Request</b>	RW	Yes	0
28	<b>DMA Channel 3 DMA No Snoop for Descriptor Write</b>	RW	Yes	0
29	<b>DMA Channel 3 DMA No Snoop for Data Write</b>	RW	Yes	0
30	<b>DMA Channel 3 In-Progress</b>	RO	No	0
31	<b>DMA Channel 3 DMA Header Logging Valid</b>	RW1C	Yes	0

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-113. 53Ch DMA Channel 3 Interrupt Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>DMA Channel 3 DMA Error Interrupt Enable</b>	RW	Yes	0
1	<b>DMA Channel 3 DMA Invalid Descriptor Interrupt Enable</b>	RW	Yes	0
2	<b>Reserved</b>	RsvdP	No	0
3	<b>DMA Channel 3 DMA Abort Done Interrupt Enable</b>	RW	Yes	0
4	<b>DMA Channel 3 DMA Graceful Pause Done Interrupt Enable</b>	RW	Yes	0
5	<b>DMA Channel 3 DMA Immediate Pause Done Interrupt Enable</b>	RW	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
16	<b>DMA Channel 3 DMA Error Interrupt Status</b>	RW1C	Yes	0
17	<b>DMA Channel 3 DMA Invalid Descriptor Interrupt Status</b>	RW1C	Yes	0
18	<b>DMA Channel 3 DMA Descriptor Done Interrupt Status</b>	RW1C	Yes	0
19	<b>DMA Channel 3 DMA Abort Done Interrupt Status</b>	RW1C	Yes	0
20	<b>DMA Channel 3 DMA Graceful Pause Done Interrupt Status</b>	RW1C	Yes	0
21	<b>DMA Channel 3 DMA Immediate Pause Done Interrupt Status</b>	RW1C	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-114. 540h DMA Channel 3 Error Header Log 0  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Error Header Log 0</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-115. 544h DMA Channel 3 Error Header Log 1  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Error Header Log 1</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-116. 548h DMA Channel 3 Error Header Log 2  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Error Header Log 2</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

**Register 18-117. 54Ch DMA Channel 3 Error Header Log 3  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>DMA Channel 3 Error Header Log 3</b> Header of TLP associated with DMA Device-Specific error.	ROS	Yes	0000_0000h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-118. 550h DMA Channel 3 Device-Specific Error Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	DMA Channel 3 Data Read Error Status	RW1CS	Yes	0
1	DMA Channel 3 Data Write Error Status	RW1CS	Yes	0
2	DMA Channel 3 Descriptor Read Error Status	RW1CS	Yes	0
3	DMA Channel 3 Descriptor Write Error Status	RW1CS	Yes	0
31:4	<i>Reserved</i>	RsvdP	No	0000_000h

**Register 18-119. 554h DMA Channel 3 Maximum Outstanding Read/Write Request  
(DMA, Upstream Port 0)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	DMA Channel 3 Maximum Outstanding Request Supported for DMA Data Read	RW	Yes	3Fh
15:6	<i>Reserved</i>	RsvdP	No	0-0h
21:16	DMA Channel 3 Maximum Outstanding Request Supported for DMA Data Write This register is invalid, and should always be programmed to 00h.	RW	Yes	00h
31:22	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 18.13 DMA Device-Specific Registers – RAM Test and Soft Error (Offsets 600h – FB0h)

This section details the Device-Specific RAM Test and Soft Error registers. [Table 18-11](#) defines the register map.

**Table 18-11. Device-Specific RAM Test and Soft Error Register Map (DMA of Upstream 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	
<a href="#">RAM Test Control</a>				600h
<a href="#">Soft Error Injection Control/Status</a>				604h
<i>Reserved</i>				608h – FB0h

**Register 18-120. 600h RAM Test Control (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
21:0	<b>RAM Test Control</b>	RW	Yes	0-0h
31:22	<i>Reserved</i>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-121. 604h Soft Error Injection Control/Status  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Descriptor RAM Set0 1-Bit Error Injection Enable</b> 1 = One bit of a symbol being transmitted by the DMA Function is corrupted	RW	Yes	0
1	<b>Descriptor RAM Set0 2-Bit Error Injection Enable</b> 1 = Two bits of a symbol being transmitted by the DMA Function is corrupted	RW	Yes	0
2	<b>Descriptor RAM Set0 Bit Field Error Injection</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RW	Yes	0
3	<b>Descriptor RAM Set1 1-Bit Error Injection Enable</b> 1 = One bit of a symbol being transmitted by the DMA Function is corrupted	RW	Yes	0
4	<b>Descriptor RAM Set1 2-Bit Error Injection Enable</b> 1 = Two bits of a symbol being transmitted by the DMA Function is corrupted	RW	Yes	0
5	<b>Descriptor RAM Set1 Bit Field Error Injection</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RW	Yes	0
15:6	<b>Reserved</b>	RsvdP	No	0-0h
23:16	<b>Descriptor RAM Set0 1-Bit Error Counter</b> Indicates the Error Count for 1-Bit ECC errors on RAM Set 0.	RO	No	00h
31:24	<b>Descriptor RAM Set1 1-Bit Error Counter</b> Indicates the Error Count for 1-Bit ECC errors on RAM Set 1.	RO	No	00h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 18.14 DMA Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

This section details the Advanced Error Reporting Extended Capability registers in DMA mode. [Table 18-12](#) defines the register map.

**Table 18-12. DMA Interface Advanced Error Reporting Extended Capability Register Map (DMA of Upstream 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		
Next Capability Offset (1F4h)	Capability Version (1h)	PCI Express Extended Capability ID (0001h)	FB4h
		Uncorrectable Error Status	FB8h
		Uncorrectable Error Mask	FBCh
		Uncorrectable Error Severity	FC0h
		Correctable Error Status	FC4h
		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 18-122. FB4h Advanced Error Reporting Extended Capability Header (DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b>	RO	Yes	0001h
19:16	<b>Capability Version</b>	RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to 1F4h, which addresses the <b>DMA Extended Capability</b> structure.	RO	Yes	1F4h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-123. FB8h Uncorrectable Error Status  
(DMA of Upstream Port)**

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>Uncorrectable Error Mask</b> register bits (DMA of Upstream Port, offset <b>FBCh</b> ).				
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>	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> <i>Reserved/Not supported</i>	RsvdP	No	0
14	<b>Completion Timeout Status</b>	RW1CS <sup>a</sup>	Yes	0
15	<b>Completer Abort Status</b> 0 = No error is detected 1 = Error is detected	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
20	<b>Unsupported Request Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
31:21	<b>Reserved</b>	RsvdP	No	0-0h

- a. When the **ECC Error Check Disable** register Software Force Error Enable bit (offset **1C8h[2]**) is Set, Type changes from RW1CS to RW.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-124. FBCh Uncorrectable Error Mask  
(DMA of Upstream Port)**

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>Uncorrectable Error Status</b> register bits (DMA of Upstream Port, offset <b>FB8h</b> ).				
3:0	<b>Reserved</b>	RsvdP	No	0h
4	<b>Data Link Protocol Error Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
5	<b>Surprise Down Error Mask</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 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
13	<b>Flow Control Protocol Error Mask</b> <i>Reserved/Not supported</i>	RsvdP	No	0
14	<b>Completion Timeout Mask</b>	RWS	Yes	0
15	<b>Completer Abort Mask</b>	RWS	Yes	0
16	<b>Unexpected Completion Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
17	<b>Receiver Overflow Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
18	<b>Malformed TLP Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
19	<b>ECRC Error Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
20	<b>Unsupported Request Error Mask</b> 0 = No mask is Set 1 = Error reporting, first error update, and Header logging are masked for this error	RWS	Yes	0
31:21	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-125. FC0h Uncorrectable Error Severity  
(DMA of Upstream Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<b>Reserved</b>	RsvdP	No	0h
4	<b>Data Link Protocol Error Severity</b> 0 = Error reported as non-fatal 1 = Error reported as fatal	RWS	Yes	1
5	<b>Surprise Down Error Severity</b>	RO	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 reported as fatal	RWS	Yes	0
13	<b>Flow Control Protocol Error Severity</b> <i>Reserved/Not supported</i>	RsvdP	No	1
14	<b>Completion Timeout Severity</b> Because the Status and Mask are both <i>reserved</i> for this bit, Severity can be ignored.	RWS	Yes	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 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 reported as fatal	RWS	Yes	0
17	<b>Receiver Overflow Severity</b> 0 = Error reported as non-fatal 1 = Error reported as fatal	RWS	Yes	1
18	<b>Malformed TLP Severity</b> 0 = Error reported as non-fatal 1 = Error 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 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 reported as fatal	RWS	Yes	0
31:21	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-126. FC4h Correctable Error Status  
(DMA of Upstream Port)**

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>Correctable Error Mask</b> register bits (DMA of Upstream Port, offset <b>FC8h</b> ).				
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
31:14	<b>Reserved</b>	RsvdP	No	0-0h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (offset **1C8h[2]**) is Set, Type changes from RW1CS to RW.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Register 18-127. FC8h Correctable Error Mask  
(DMA of Upstream Port)**

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>Correctable Error Status</b> register bits (DMA of Upstream Port, offset <b>FC4h</b> ).				
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
31:14	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Register 18-128. FCCh Advanced Error Capabilities and Control  
(DMA of Upstream Port)**

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 <b>FB8h</b> ).	ROS	No	1Fh
5	<b>ECRC Generation Capable</b> 0 = ECRC generation is not supported 1 = ECRC generation is supported, but must be enabled	RO	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	RO	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

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Register 18-129. FD0h Header Log 0**  
**(DMA of Upstream Port)**

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	0-0h

**Register 18-130. FD4h Header Log 1**  
**(DMA of Upstream Port)**

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	0-0h

**Register 18-131. FD8h Header Log 2**  
**(DMA of Upstream Port)**

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	0-0h

**Register 18-132. FDCh Header Log 3**  
**(DMA of Upstream Port)**

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	0-0h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3



## Chapter 19 Dual Clocking Support

### 19.1 Introduction

The Dual Clocking feature allows the PEX 8619 to be used in a Spread Spectrum Clocking (SSC) environment and communicate with devices that are operating in a Constant Frequency Clocking (CFC) environment. This capability is necessary for systems in which the upstream Port connects to the SSC source, while downstream Ports connect to Links running on CFC domains.

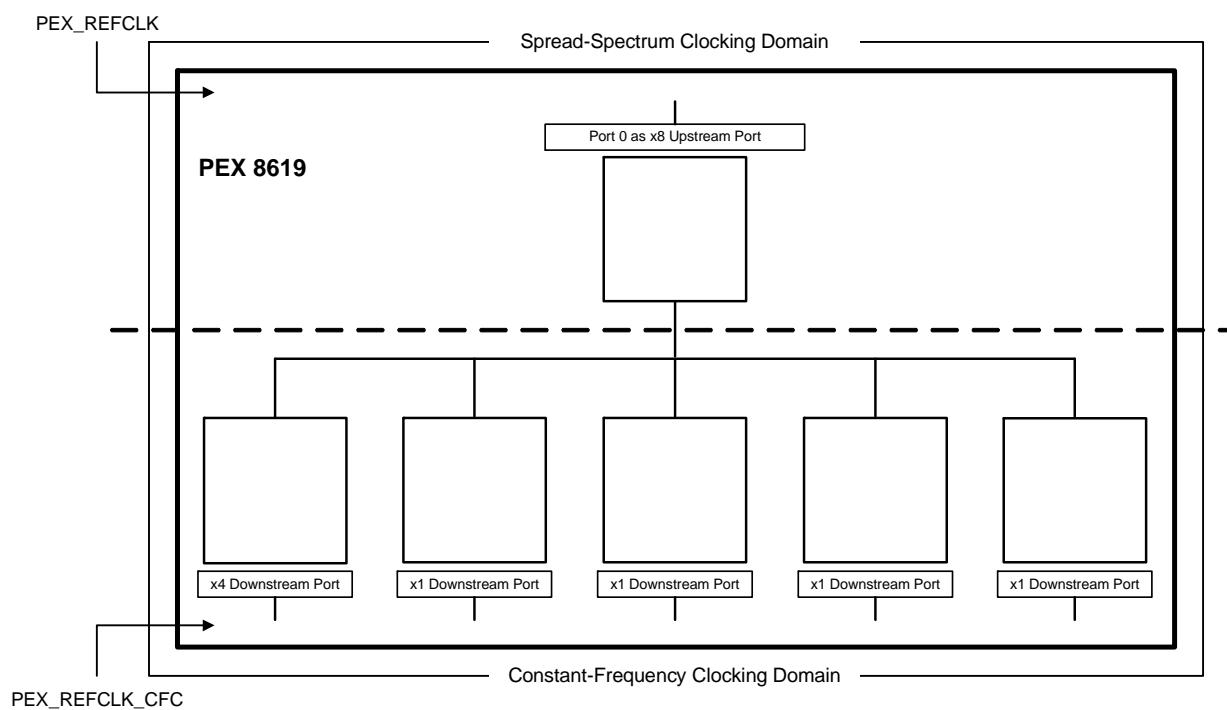
The SSC isolated Port does not have to be the upstream Port; however, it does need to be Port 0, because all the SSC resources are located in Port 0. Port 0 must be connected to an SSC domain, and the other Ports must be connected to a CFC domain, as illustrated in [Figure 19-1](#).

Most systems support SSC; however, the *PCI Express Base r2.0* specifies that REFCLKs of two interconnected devices must be within a  $\pm 300$  ppm frequency tolerance of one another.

The *PCI Express Base r2.0* specifies that the data rate of two connected Ports must be within  $\pm 300$  ppm from one another. The *PCI Express Base r2.0* also specifies that the data rate can be SSC modulation, which is defined as a  $+0/-5,000$  ppm modulation from the base frequency. This essentially requires that if one device is using a SSC clock, then the other device must use the same clock, to be able to track the frequency modulation. Therefore, a device on a system that supports SSC clocking cannot be connected to a device on another system that does not use SSC. If a system supports SSC, all devices in the PCI Express interconnect must have the same Reference Clock source, to allow traffic to communicate between them. However, this is not practical in the case of chassis-to-chassis or blade-to-blade PCI Express traffic. If each chassis or blade has its own SSC source, special clock isolation circuitry is required to enable such configurations to communicate between themselves. For this reason, the PEX 8619 supports two REFCLK inputs that allow traffic to move from the SSC clock domain on the upstream Port to a CFC clock domain on the downstream Ports, thereby allowing two different chassis or blades to communicate with one another.

The SSC domain runs at a frequency between 99.5 and 100 MHz. The CFC domain runs at a frequency of 100 MHz. Because the SSC and CFC domain average frequencies are not identical, the PEX 8619 provides buffering and flow control to ensure that data does not overflow nor underflow as it passes from one clock domain to the other. Dual Clocking support is provided in fan-in/fan-out configurations when Port 0 is programmed to a x8 or x4 Link width.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 19-1. PEX 8619 in a Dual Clock System****NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 19.2 Dual Clocking Operation

Dual Clocking is enabled when the following conditions are met:

- **STRAP\_SSC\_ISO\_ENABLE#** input is pulled or tied Low to **VSS** (GND)
- Port 0's Link width is x8 or x4

When Dual Clocking is enabled, the REFCLK balls are mapped as follows:

- **PEX\_REFCLKn** and **PEX\_REFCLKp** signal balls become the SSC domain Clock signals
- **PEX\_REFCLK\_CFCn** and **PEX\_REFCLK\_CFCp** signal balls become the CFC domain Clock signals

If the system design does not use Dual Clocking, the **STRAP\_SSC\_ISO\_ENABLE#** input must be pulled or tied High to **VDD25**, and the **PEX\_REFCLK\_CFCn** and **PEX\_REFCLK\_CFCp** signals must remain floating. If the PEX 8619 uses SSC on its upstream Port, or the upstream Port is not configured to Port 0, the **PEX\_REFCLKn** and **PEX\_REFCLKp** signals must be shared by all downstream devices. If the PEX 8619 uses a Constant Frequency source on its upstream Port, each downstream device can use an independent CFC REFCLK source, provided that it meets the *PCI Express Base r2.0*-defined PCI Express frequency tolerance of  $\pm 300$  ppm.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 20 Test and Debug

### 20.1 Introduction

This chapter describes the following test- and debug-related information:

- Physical Layer Loopback Operation
- User Test Pattern
- Pseudo-Random Bit Sequence
- Using the Even/Odd Port SerDes Quad x Diagnostic Data Registers
- Pseudo-Random and Bit-Pattern Generation
- PHY Testability Features
- JTAG Interface
- Lane Good Status LEDs

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

## 20.2 Physical Layer Loopback Operation

### 20.2.1 Overview

Physical Layer (PHY) Loopback functions are used to test the SerDes in the PEX 8619, connections between devices, and SerDes of external devices, as well as various PEX 8619 and external digital logic. The PEX 8619 supports four types of Loopback operations, as described in [Table 20-1](#). Additional information regarding each type is provided in the sections that follow.

**Table 20-1. Loopback Operations**

Operation	Description
Analog Loopback Master Mode	This mode depends upon an external device or passive connection ( <i>such as a cable</i> ) to loopback the transmitted data to the PEX 8619, 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, so that SKIP Ordered-Sets are not inserted. A device's re-transmitted Receive data must be sent back to the Master, synchronous to the Master's Transmit Reference Clock. <i>That is</i> , the Slave device re-serializes the Transmit data, using the recovered clock from the received data. In that mode, the PRBS generator and checker should be used to create and check the data pattern.
Digital Loopback Master Mode	This mode depends upon an external device to loopback the transmitted data that includes at least its Elastic buffer in the Loopback data path, allowing for reliable loopback testing, in case the two devices have asynchronous Reference Clock sources with Parts per Million (PPM) offsets. The Master's pattern generator inserts SKIP Ordered-Sets at regular intervals, and its received data checker can handle PPM offset clock compensation, by way of SKIP symbol addition or deletion. The PEX 8619 provides a User Test Pattern generator and checker that can be used for Digital loopback testing.
Analog Loopback Slave Mode	The PEX 8619 enters this mode when an external device transmits Training Sets with the <i>Loopback Training Control Bit Set</i> and the <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <i>23Ch[21]</i> ) is Set. Another way to unconditionally force the Slave into Analog loopback is described later in <a href="#">Section 20.2.4</a> . While in this mode, the received data is looped back from the SerDes 10-bit Receive interface to the 10-bit Transmit interface. Internal to the SerDes, the serial-to-parallel and parallel-to-serial converters are included in the Loopback data path. The re-serialized data is transmitted back to the Master device synchronous to that Master's Reference Clock. This is because the recovered clock is fed back around to the Transmit interface and used as the Tx clock.
Digital Loopback Slave Mode	The PEX 8619 enters this mode when an external device transmits Training Sets with the <i>Loopback Training Control Bit Set</i> and the <b>Common Physical Layer Safety and Test</b> register <i>Analog Loopback Enable</i> bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset <i>23Ch[21]</i> ) is Cleared. This is the default Loopback mode for the LTSSM Slave <i>Loopback.Active</i> state. In this mode, the data is looped back at the 8-bit level, which includes the PEX 8619's Elastic buffer, 8b/10b decoder, and 8b/10b encoder in the Slave Loopback data path. Asynchronous clock compensation can occur in the Elastic buffer through SKIP symbol addition or deletion, depending upon clock PPM offsets and fill threshold decoding. The Master data pattern checker must be able to handle the presence of SKIP Ordered-Sets and variations in their contents.

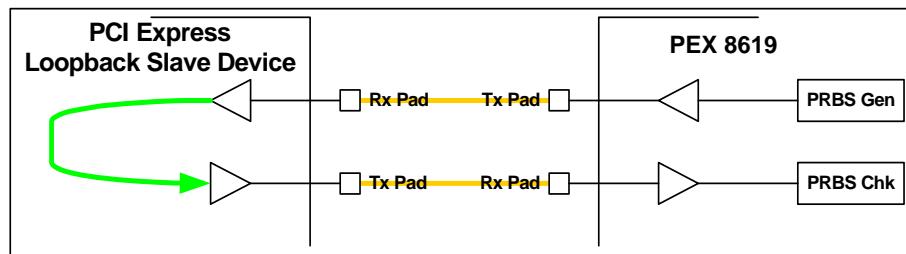
NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

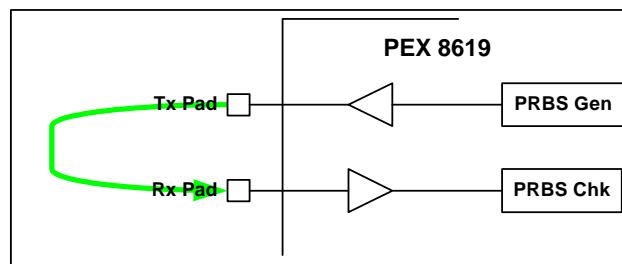
## 20.2.2 Analog Loopback Master Mode

Analog Loopback Master mode is typically used for Analog Far-End testing (refer to [Figure 20-1](#)), with a shallow Loopback path Slave device, to determine overall Bit Error rates. However, it can also be used for passive external serial loopback 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 20-2](#). 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 – refer to [Section 20.3](#) for details.

**Figure 20-1. Analog Far-End Loopback**



**Figure 20-2. Cable Loopback**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

### 20.2.2.1 Initiating Far-End Analog Operations in PEX 8619 Master Devices

**Note:** *Initiating a Master Loopback operation on an upstream Port can cause a Deadlock condition to occur, unless an I<sup>2</sup>C Slave interface is used to write and read Configuration Space register bits instead of writing them through upstream Port Configuration transactions. Therefore, it is recommended to restrict Analog Master loopback testing to downstream Ports when external devices are used.*

One way to test Master Analog loopback with passive cables is to have an upstream Port connected to a Root Complex, for Configuration Write/Read transactions that are used to Set and monitor the key device register bits. In that case, only downstream Ports would be test-capable, to avoid potential Deadlock conditions on the upstream Port. Alternatively, an I<sup>2</sup>C Slave interface and Rapid Development Kit (RDK) software could be used to write to or read from the registers. This makes any Port testable. The user has the option of attaching one or more cables to the appropriate high-speed Tx and Rx differential pairs that belong to the Ports being tested.

Loopback cables can be attached before or after a standard power-up initialization sequence. If the cables are attached before power-up, use a serial EEPROM to program the **Even/Odd Port Physical Layer Command/Status** register *Port x Loopback Command* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 220h[28, 24, 20, 16, 12, 8, 4, 0], and Odd Ports, offset 224h[28, 24, 20, 16, 12, 8, 4, 0]) for the Port being tested. The Port's *Port x Loopback Command* bit arms the Port to enter the *Master Loopback.Entry* state. When written from a serial EEPROM, the bit's assertion is present before the Ports begin Link training. In that case, the Ports directly transition to the LTSSM *Loopback* state from the LTSSM *Configuration* state. The LTSSM exits the *Polling* state and enters the *Configuration.LinkWidth.Start* state, then immediately transitions to the *Master Loopback.Entry* state.

At this point, users can sample the Port's **Even/Odd Port Physical Layer Command/Status** register *Port x Ready as Loopback Master* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 220h[31, 27, 23, 19, 15, 11, 7, 3], and Odd Ports, offset 224h[31, 27, 23, 19, 15, 11, 7, 3]), to determine whether the bit is Set, which indicates that the Master has reached the LTSSM *Loopback.Active* state. At this time, the PRBS engine can be enabled, by Setting the **Even/Odd Port User Test Pattern Delay** register *SerDes x PRBS Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[31:24], and Odd Ports, offset 25Ch[31:24]) associated with the SerDes assigned to the Port being tested.

The PRBS Receive data checker first synchronizes the de-serialized parallel data words from the returned pattern with a reference PRBS pattern generator. At this point, users should check the Lane synchronization status in the **Even/Odd Port Loopback Master Status/Control** register *PRBS Pattern Sync Status Device Lane x* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset B80h[23:16], and Odd Ports, offset B84h[23:16]). If there is no synchronization, there is likely a physical connection problem. Once synchronized, the PRBS checker looks for errors, on a continuous basis. Any errors detected are logged in one or more of the **Even/Odd Port SerDes Quad x Diagnostic Data** register RO bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 240h through 24Ch, bits [30 and 23:0]). The errors can be retrieved, by reading the appropriate bit.

If the *Port x Loopback Command* bits are not Set through the serial EEPROM, the Ports' Loopback Training Sets can be used to cause the Ports to linkup, by way of a Configuration cross-link track, resulting with the Ports being in L0 Link PM state. This linkup of a Port, in response to its own Training Sets, only works if the Port's **Even/Odd Port Loopback Master Status/Control** register *Port x External Loopback Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset B80h[15:8], and Odd Ports, offset B84h[15:8]) is Set by serial EEPROM. Once the Port is in the L0 Link PM state, Configuration Space register programming can then be performed manually, to invoke a Master Loopback operation.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

Once the Ports linkup, users can direct the Ports into an Analog Loopback Master condition, by writing the *Port x Loopback Command* bit(s), through the upstream Port and/or I<sup>2</sup>C Slave interface. However, this is not sufficient to initiate the LTSSM transition from the L0 Link PM state, to the *Loopback* state. The Link must pass through a *Recovery* substate, before the *Port x Loopback Command* bits can be sampled and allow the LTSSM to pass through the *Recovery* state to the *Loopback* state. To cause the Port to enter the *Recovery* state, users must Set the Port's **Link Control** register *Retrain Link* bit (offset 78h[5]). At this point, users should monitor the Port's *Port x Ready as Loopback Master* bit(s), and when Set, the PRBS engine(s) can be enabled, as previously described.

If loopback cables are attached after the device powers up, then those Ports whose Lanes are floating unconnected did not detect Receivers. Therefore, those Ports are not trained up to the L0 Link PM state.

If the Port's *Port x Loopback Command* and *Port x External Loopback Enable* bit(s) for the downstream Ports to be tested are written *before* the cables are attached, then once cabled, there is Receiver detection, the Port(s) go through Link training, and then exit the LTSSM *Configuration* state and directly enter the *Loopback* state.

However, if the Port's *Port x Loopback Command* and *Port x External Loopback Enable* bit(s) are Set *after* the cables are attached, the Ports do not recognize their own Training Sets and will likely cycle back and forth between *Configuration* and *Detect*. Therefore, users must at least Set the *Port x External Loopback Enable* bit for the Ports being tested, by way of serial EEPROM, if the PEX 8619 is powered up before the cables are attached. Users can then program the Port's *Port x Loopback Command* bit(s). In addition to this, a forced retrain is also needed, to enter into the *Loopback* state through the *Recovery* state, as previously described.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.2.3 Digital Loopback Master Mode

The only difference between Analog and Digital Loopback Master modes is that the external device is assumed to have at least an Elastic buffer in the Loopback data path. Because of this, SKIP Ordered-Sets must be included in the test data pattern, which precludes use of the PRBS engine.

[Figure 20-3](#) illustrates a Far-End Digital Loopback Master connection and data path.

The PEX 8619 provides a User Test Pattern engine on a per-Lane basis, for Digital Far-End Loopback testing. The user pattern itself, however, is common to all Lanes where it is enabled. Details on the use of the User Test Pattern registers and controls are described later in [Section 20.5](#).

What is important to note about the data path (not shown in [Figure 20-3](#)) is that the pattern generators and checkers in the PEX 8619 Digital Loopback Master have 8/10b encode, 10b/8b decode, and Elastic buffers included in the Tx/Rx path. The scramblers and de-scramblers are disabled. Therefore, the Digital Loopback Slave device must not scramble the returning data. The 10-bit data can be decoded to 8-bit, and encoded back to 10-bit as an option, and will not affect the UTP pattern checker in the PEX 8619, unless there is a coding error.

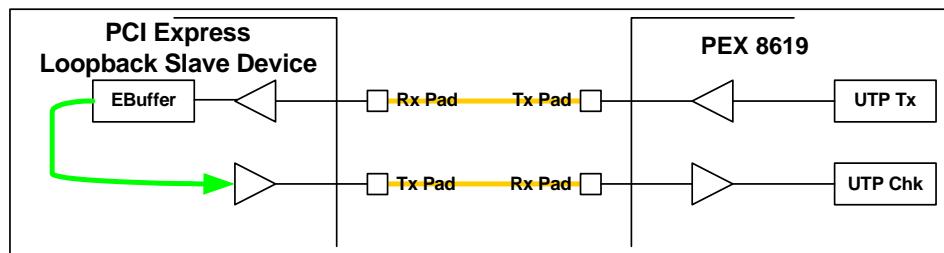
Digital Loopback Master mode is established by either programming method previously described in [Section 20.2.2](#) for Analog Loopback Master mode. The Port's **Even/Odd Port Physical Layer Command/Status** register *Port x Loopback Command* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [220h\[28, 24, 20, 16, 12, 8, 4, 0\]](#), and Odd Ports, offset [224h\[28, 24, 20, 16, 12, 8, 4, 0\]](#)) can be Set with a serial EEPROM, causing Loopback to be entered directly from the LTSSM Configuration state. Otherwise, the Port's *Port x Loopback Command* bit can be Set after linkup, and then the Port's **Link Control** register **Retrain Link** bit (offset [78h\[5\]](#)) can be used to move the Port to the *Loopback* state, through the LTSSM Recovery state.

Once Digital Loopback Master mode is established, Configuration Space register Writes are used to establish a User Test Pattern transmission, as well as error checking, which are described later in [Section 20.3](#).

The UTP is multiplexed, unconditionally, onto the Transmit data path, upon Setting one or more of the **Even/Odd Port User Test Pattern Delay** register *SerDes x User Test Pattern Enable* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [258h\[23:16\]](#), and Odd Ports, offset [25Ch\[23:16\]](#)).

**Note:** *It is important to verify that the LTSSM is in a Master Loopback.Active state, before writing 1 to the SerDes x User Test Pattern Enable bits. Therefore, do not use the serial EEPROM to Set the SerDes x User Test Pattern Enable bits. (Refer to [Section 20.4](#) for details.)*

**Figure 20-3. Digital Far-End Loopback**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.2.4 Analog Loopback Slave Mode

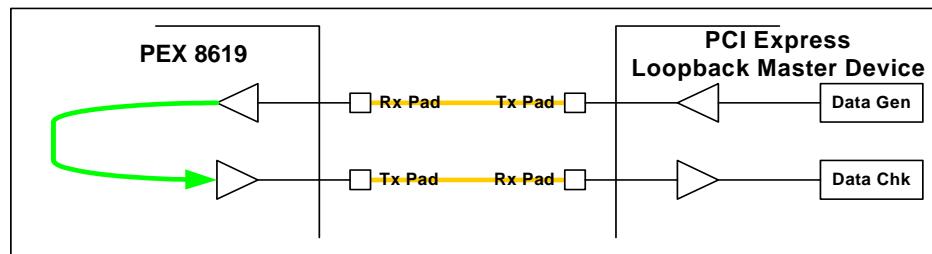
The PEX 8619 becomes an Analog Loopback Slave (as illustrated in Figure 20-4) if it receives Training Sets with the *Loopback* Training Control Bit Set while the **Common Physical Layer Safety and Test** register *Analog Loopback Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]) is Set. It is recommended that the *Analog Loopback Enable* bit be Set first, before bringing the PEX 8619 into an LTSSM Slave *Loopback.Active* state.

As previously described, Analog Loopback does not have only pure analog circuitry in the Slave's data path. While in this mode, the received data and recovered clock are looped back from the SerDes Parallel Receive Data interface to the Parallel Transmit interface. Internal to the SerDes, the serial-to-parallel and parallel-to-serial converters are included in the Loopback data path. The re-serialized data is transmitted back to the Master device, synchronous to that Master's Reference Clock.

The multiplexing control that enables Parallel data from the Receive path, directly back to the Transmit path, is held off from asserting until the Slave reaches the *Loopback.Active* state. Then, the Parallel Recovered data and clock are multiplexed back into the SerDes Parallel Transmit interface. That multiplexer remains effective until the PEX 8619 Loopback Slave exits the *Loopback.Active* state. There are alternate ways to transition out of the Slave *Loopback.Active* state:

- If the Loopback is operating at Gen 2 (5.0 GT/s) Link speed, receipt of four consecutive Electrical Idle Ordered-Sets (EIOS) causes a Loopback exit
- If the Link is operating at Gen 1 (2.5 GT/s) Link speed, then receipt of a single EIOS, or detection of Electrical Idle entry, causes an exit
- If the Slave device appears to be “stuck” in the *Loopback.Active* state, toggling of the Port’s **Even/Odd Port Physical Layer Command/Status** register *Port x Loopback Command* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 220h[28, 24, 20, 16, 12, 8, 4, 0], and Odd Ports, offset 224h[28, 24, 20, 16, 12, 8, 4, 0]), from 0 to 1 to 0, breaks the LTSSM out of Loopback Slave operation

**Figure 20-4. Analog Loopback Slave Mode**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

Analog Loopback Slave mode is most suitable for a PRBS test pattern. However, because the PEX 8619 includes only the SerDes in the Loopback data path, and the Transmit data is clocked out at the recovered clock frequency, the Master can include SKIP Ordered-Sets in its data pattern, regardless of whether the system uses synchronous or asynchronous clocking, as long as it can tolerate the presence of SKIP Ordered-Sets in the data pattern. In this case, the Master sees its own SKIP Ordered-Sets returned to it, at the same intervals and positions in the data pattern.

If the Master device is not capable of bringing the PEX 8619 to a Slave *Loopback.Active* state (such as a Bit Error Rate Tester (BERT) as the Master) through the LTSSM state transition arcs previously described, there is a way to unconditionally force the Slave into Analog Loopback Slave mode, through device Configuration Space register Writes to the appropriate bit states. The *BERT Loopback Path Enable* bits (**Even/Odd Port Physical Layer Test** register *Lane x Parallel Loopback Path Enable* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [228h\[31:24\]](#), and Odd Ports, offset [22Ch\[31:24\]](#))) are first Set for all device Lanes associated with the Port being tested.

Next, the **Common Physical Layer Safety and Test** register *Analog Loopback Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [23Ch\[21\]](#)) is Set. The Lanes enabled by the BERT's *Lane x Parallel Loopback Path Enable* bits immediately go into an Analog Loopback path mode, *regardless of the Slave's current LTSSM state*.

Concurrently Setting the *Analog Loopback Enable* and BERT's *Lane x Parallel Loopback Path Enable* bits, for all Lanes of the Port being tested, changes the Loopback data path, as described; however, that does not guarantee that the SerDes Transmitters are powered on and operating at the correct speed. Other PHY Safety bits can be used to ensure that the SerDes are powered up and ready to be placed into Analog Loopback Slave mode. The **Even/Odd Port Disable/Quiet/Test Pattern Rate** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets [230h](#) and [234h](#)) contains useful bits for controlling loopback operations:

- *Disable Port x* bits – Even Ports, offset [230h\[7:0\]](#), and Odd Ports, offset [234h\[7:0\]](#)
- *Hold Port x Quiet* bits – Even Ports, offset [230h\[15:8\]](#), and Odd Ports, offset [234h\[15:8\]](#)
- *Port x Test Pattern x Rate* bits – Even Ports, offset [230h\[23:16\]](#), and Odd Ports, offset [234h\[23:16\]](#)

The Port's *Hold Port x Quiet* bit holds the Port in the *Detect.Quiet* state once it enters that state, and does not allow the LTSSM to advance. The Port's *Hold Port x Quiet* bits also keep the SerDes Transmitters and Receivers powered on, as long as the Port's *Disable Port x* bit is not Set. The Port's *Port x Test Pattern x Rate* bit, if Set, forces the Port's SerDes to shift their Link speed to Gen 2 (5.0 GT/s) **if** the Port's *Hold Port x Quiet* bit is also Set.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

When forcing the PEX 8619 into Analog Loopback Slave mode with a BERT attached, the Port's LTSSM looks at whatever the BERT is transmitting on the attached Lanes. Because the BERT does not transmit Training Sets, the LTSSM detects Receivers, goes to the *Polling* state, times out, and then returns to the *Detect* state to try again. The LTSSM should remain in the *Detect.Quiet* state once it returns to that state. The Port's *Hold Port x Quiet* bit, therefore, should be Set, to hold the PEX 8619 in a stable LTSSM *Detect.Quiet* state, that keeps the SerDes powered on and prevents additional state transitions. If it is necessary for the BERT to test the SerDes at the Gen 2 rate, Set the Port's *Port x Test Pattern x Rate* bit before Setting its *Hold Port x Quiet* bit.

*To Set the necessary bits prior to BERT testing, for the Port being tested:*

1. Use I<sup>2</sup>C to Set the Port's **Even/Odd Port Quiet** register *Hold Port x Quiet* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset **230h[15:8]**, and Odd Ports, offset **234h[15:8]**).
2. If Analog Loopback Slave mode must operate at the Gen 2 rate (5.0 GT/s), Set the Port's **Even/Odd Port Test Pattern Rate** register *Port x Test Pattern x Rate* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset **230h[23:16]**, and Odd Ports, offset **234h[23:16]**).
3. Set the BERT's **Even/Odd Port Physical Layer Test** register *Lane x Parallel Loopback Path Enable* bit(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset **228h[31:24]**, and Odd Ports, offset **22Ch[31:24]**).
4. Set the **Common Physical Layer Safety and Test** register **Analog Loopback Enable** bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **23Ch[21]**).

The Slave device should now be in Analog Loopback Slave, properly powered, and at the correct Link speed for BERT testing.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.2.5 Digital Loopback Slave Mode

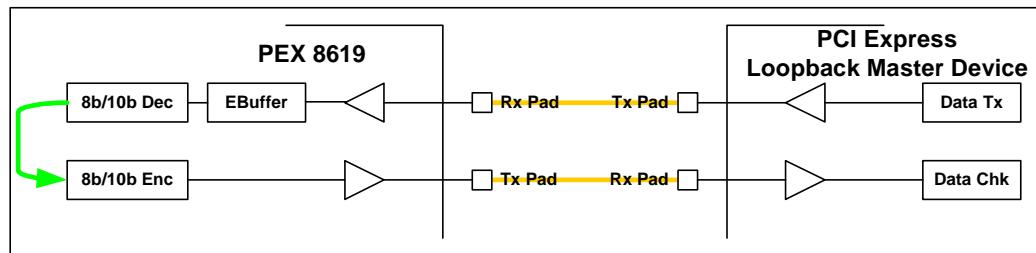
When a PEX 8619 Port is in the LTSSM Slave *Loopback.Active* state, it automatically becomes a Digital Loopback Slave, by default. The Port enters this state after it receives Training Sets with the *Loopback* Training Control Bit Set.

When a PEX 8619 Port is a Digital Loopback Slave, it includes the Elastic buffer, 8b/10b decoder, and 8b/10b encoder in the Loopback data path. The Loopback Master must provide the test data pattern and data pattern checker (*such as a PEX 8619 User Test Pattern*). The Loopback Master must also transmit SKIP Ordered-Sets with the data pattern. Depending upon the PEX 8619 Reference Clock source's PPM offset, the PEX 8619 Digital Loopback Slave's Elastic buffers can compensate for the offset, by returning more or fewer SKIP symbols than the PEX 8619 received from the Master. Therefore, the Master's data pattern checker must make provisions for this when decoding for errors.

This mode is *not* suitable for a PRBS pattern as transmitted from the Master, because neither device can compensate for Reference Clock offset differences, should they exist.

Unlike Analog Loopback Slave mode (described in [Section 20.2.4](#)), there is no way to unconditionally force the Loopback path into Digital Loopback Slave mode, through the use of PHY-related register bits – the Slave must be brought into the mode by a Master-connected device, through standard LTSSM tracks.

**Figure 20-5. Digital Loopback Slave Mode**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.3 User Test Pattern

The PEX 8619 provides a User Test Pattern (UTP) Transmit and Receive data checker, for Digital Far-End Loopback testing. (Refer to [Figure 20-3](#).) After LTSSM Loopback Master mode is established, Configuration Writes are used to fill the **Physical Layer User Test Pattern, Bytes x through y** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets [210h](#) through [21Ch](#)). One or more **Even/Odd Port User Test Pattern Delay** register *SerDes x User Test Pattern Enable* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [258h\[23:16\]](#), and Odd Ports, offset [25Ch\[23:16\]](#)) are used to start the UTP transmission, on the Lanes assigned to each bit. The UTP logic assumes that there is asynchronous clocking between the PEX 8619 Loopback Master and the connected Slave device. Therefore, the expectation is that there is at least a clock-crossing boundary in the Slave device's Loopback data path (*such as an Elastic buffer*). SKIP Ordered-Sets are inserted into the user's test data pattern, at the nearest data pattern boundary according to the programmed SKIP interval. That interval is determined by the **Common Physical Layer Safety and Test** register *SKIP Ordered-Set Interval* field (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [23Ch\[19:8\]](#)) value. The default interval is 1,180 symbol times.

The Test Pattern checker ignores SKIP Ordered-Sets returned by the Loopback Slave, because the quantity of SKIP symbols received can be different from the quantity transmitted. All other data is compared to the transmitted data, and errors are logged in the **Even/Odd Port SerDes Quad x Diagnostic Data** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets [240h](#) through [24Ch](#)).

The 16-byte UTP is loaded into the **Physical Layer User Test Pattern, Bytes x through y** registers. The pattern is common to all Lanes. Prior to transmission, the 8b/10b encoder converts the 16 bytes to 10-bit encoded data. Pattern bytes only go out as control symbols (k-bit set), if the Lane's **Common Physical Layer Command/Status** register *User Test Pattern K-Code Flag* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset [238h\[31:16\]](#)) is Set.

**Caution:** *Use caution when Setting User Test Pattern K-Code Flag bits, because UTP logic does not check the validity of Control characters.*

The UTP Transmitter logic does not immediately transmit the UTP bytes upon being enabled – a fixed, 4-byte sync pattern (5243h) is continuously transmitted first. The sync word detection validates the physical Loopback wiring and connected device Loopback path, to qualify the UTP transmission's initiation. The sync DWord allows the Pattern Checking logic to determine the starting boundary of the received pattern byte sequence. Sync detection also enables Received Data error checking and logging. Unfortunately, there are no sync-acquired status bits in the Physical Layer registers, like there are for PRBS; therefore, it is not possible to verify that the sync pattern has been detected. However, a single UTP Error Count is logged if the sync pattern is not detected within 256 ns from the initial transmission of the sync DWord. Therefore, a single Error Count may, or may not, indicate the absence of received sync data (*for example*, a good sync could have been followed by a single bit error).

**Notes:** *There are no explicit Control bits for deliberately injecting UTP errors into the transmission, to test the error checking ability. However, one way of testing the ability is to write a test pattern byte to a different value after the transmission has started. That usually causes a temporary unequal boundary condition, which will log an error. While not guaranteed to inject an error, this method is useful for testing error checking ability.*

*A UTP is not recommended for Master mode far-end cable testing, especially when initiated by way of serial EEPROM from a power-up sequence. If a UTP is enabled and looped back before Link training begins, the symbol framers will not have seen any COM symbols, and the true 10-bit symbol boundaries are unknown. The framer requires three COMs in a row, in the same bit position, to achieve symbol lock. Neither the sync pattern, nor the user pattern, would be detected in this case, and the test is certain to fail.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

In addition to the 16-byte pattern registers, the UTP is enabled on a per-Lane basis, by Setting the **Even/Odd Port User Test Pattern Delay** register *SerDes x User Test Pattern Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset [258h\[23:16\]](#), and Odd Ports, offset [25Ch\[23:16\]](#)), for the SerDes associated with the Port being tested.

*Note: The UTP is unconditionally multiplexed onto the Transmit data path, upon Setting the SerDes x User Test Pattern Enable bits. Therefore, it is necessary to verify that the LTSSM is in an LTSSM Master Loopback.Active state before writing a value of 1 to those Enable bits. Do not use a serial EEPROM to Set the SerDes x User Test Pattern Enable bits.*

UTP testing results can be monitored in one of the **Even/Odd Port SerDes Quad x Diagnostic Data** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets [240h](#) through [24Ch](#)). Each register can be used to examine the Error Count in the *UTP/PRBS Error Counter* field [[23:16](#)], and expected/actual data of the first failing byte (fields [[7:0](#) and [15:8](#)], respectively). The *Status* bits are on a per-Lane basis associated with a SerDes quad. **The important field in these registers is the SerDes Diagnostic Data Select field [25:24].** When the Lane code for the quad is written to that field, the UTP status for that Lane appears in the *Status* fields. Bit [30](#) of the register (*PRBS Counter/-UTP Counter*) is a pattern type indicator, and is Cleared when UTP is enabled for a Lane.

*Notes:* Use of the Diagnostic registers is explained in [Section 20.5](#).

***The UTP and PRBS Enables are mutually exclusive, and must not be Set concurrently. If both Enables are simultaneously Set, the resulting operation is undefined. The UTP/PRBS Error Counter field continues to count, until it saturates at 255. To clear the Counter and allow it to begin logging errors again, write all zeros (0) to that field. Alternatively, the Counter status is Cleared if the SerDes x User Test Pattern Enable bit for that Lane is Cleared, and then Set again.***

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.4

## Pseudo-Random Bit Sequence

A Pseudo-Random Bit Sequence (PRBS) generator and checker are useful as a diagnostic/debugging tool, and for measuring short- or long-term bit error rates in PCI Express systems. The PEX 8619 also uses a specially enabled power-up self-test that runs after reset, as a wafer sort test for use on automated test equipment. PRBS pattern generators and checkers reside within the SerDes\_rclk\_blk modules, because they transmit and receive 10- or 20-bit data directly to/from the SerDes quads. Locating them in the modules helps ensure tight timing and short trace length on SerDes Tx and Rx parallel data.

The PEX 8619 PRBS logic is enabled by one or more of the **Even/Odd Port User Test Pattern Delay** register *SerDes x PRBS Enable* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[31:24], and Odd Ports, offset 25Ch[31:24]), for the SerDes associated with the Port being tested. Prior to enabling PRBS, an externally connected PCI Express device must be in an LTSSM Slave *Loopback.Active* state. Furthermore, the reference clocking between the two devices must be synchronous. (*That is*, the returning PRBS pattern must have its transmission clock source synchronous to the PEX 8619 Reference Clock.) The PEX 8619 PRBS pattern generator does not insert any SKIP Ordered-Sets, and, if the Slave device inserts SKIP Ordered-Sets into the returning pattern, they cannot be ignored by the PRBS checker (it causes an error). Alternatively, the PRBS pattern can be used to test an external cable Loopback, after the correct LTSSM Master *Loopback.Active* state is reached, as described in [Section 20.2.2](#).

After a PEX 8619 Lane's PRBS engine is enabled, the PRBS engine immediately begins to transmit the PRBS pattern on that Lane. No 8b/10b encoding is performed. The PRBS pattern generator produces 10- or 20-bit symbols on every Clock cycle, depending upon the current Link speed. The symbols are written directly into the SerDes Tx data Port, for immediate transmission.

The PRBS Receive Data Checking logic first synchronizes the de-serialized 10- or 20-bit Parallel Data symbols from the SerDes Rx data Port, using a reference PRBS pattern generator. After pattern synchronization is achieved, the Receive data checker begins comparing the Rx data symbols on a continuous basis, to discover any mismatch between a symbol's expected and received values.

*Note: Error checking cannot begin until synchronization is achieved; therefore, it is important to monitor the pattern synchronization status, before checking the error status. Synchronization status is available in the **Even/Odd Port Loopback Master Status/Control** register PRBS Pattern Sync Status Device Lane x bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset B80h[23:16], and Odd Ports, offset B84h[23:16]). This status should always be checked. If there is no synchronization, there is likely a physical connection problem. Any errors detected are logged in one or more of the **Even/Odd Port SerDes Quad x Diagnostic Data** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 240h through 24Ch). Use of these registers is explained in [Section 20.5](#).*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

It is possible to inject errors into a PRBS pattern while it is transmitting on a per-Lane basis, for testing the ability to find and count errors, by programming the **Even/Odd Port SerDes Quad x Diagnostic Data** register *PRBS Error Injection* field(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **240h** through **24Ch**, field [29:26]) to 1h. When enabled, an error is injected, by corrupting a single bit in one PRBS pattern symbol once every 127 words.

PRBS testing results can be monitored in one of the **Even/Odd Port SerDes Quad x Diagnostic Data** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **240h** through **24Ch**). Each register can be used to examine the Error Count in the *UTP/PRBS Error Counter* field [23:16], and expected/actual data of the first failing byte (fields [7:0] and **15:8**, respectively). The *Status* bits are on a per-Lane basis associated with a SerDes quad. **The important field in these registers is the *SerDes Diagnostic Data Select* field [25:24].** When the Lane code for the quad is written to that field, the PRBS status for that Lane appears in the *Status* fields. Bit **30** of the register (*PRBS Counter/-UTP Counter*) is a pattern type indicator, and is Set when PRBS is enabled for a Lane.

**Notes:** Use of the Diagnostic registers is explained in [Section 20.5](#).

***The UTP and PRBS Enables are mutually exclusive, and must not be Set concurrently. If both Enables are simultaneously Set, the resulting operation is undefined. The UTP/PRBS Error Counter field continues to count, until it saturates at 255. To Clear the Counter and allow it to begin logging errors again, write all zeros (0) to that field. Alternatively, the Counter status is Cleared if the SerDes x User Test Pattern Enable bit for that Lane is Cleared, and then Set again.***

*The PRBS Error Count does not necessarily represent a true Bit Error rate. The PRBS checker detects one or more mismatched bits in each examined symbol, on a symbol-per-core-clock basis. Therefore, the Error Counter advances one count for every symbol mismatch, regardless of how many bits are in error for that failing symbol.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.4.1 Physical Layer Built-In Self-Test

The PEX 8619 also uses the PRBS engines in a special power-up self-test of all SerDes. This is called *Physical Layer Built-In Self-Test (PHY BIST)*, and is intended for production wafer sorting, to screen out die that have basic SerDes Rx/Tx data/clock circuitry faults. The test uses the Internal Serial Loopback mode, whereby the serial Tx data is looped back to the serial Rx data path prior to the differential pad buffer cells. PHY BIST is enabled with a unique code (5h; LHLH) applied to the **STRAP\_TESTMODE[3:0]** device Ports, and runs at Gen 2 (5.0 GT/s) Link speed.

**Note:** If the SerDes must be brought up to 5.0 GT/s, it must be done without bringing the Link up in the same way that the Functional logic would bring the SerDes up. Instead, the PHY BIST uses the PIPE interface Rate signaling sequence, to change the speed before it starts up the PRBS generators, and then checks for Sync and Data errors.

After the PHY Reset is released, the PHY BIST logic starts up if PHY BIST is enabled. If enabled, Internal Serial Loopback mode, as well as the PRBS engines for all device Lanes, are also enabled. PRBS Loopback pattern generation runs for 16 ms, and then stops. Any errors detected are logged in a special status, that is enabled onto the GPIO[6:0] balls. **Table 20-2** lists the GPIO[6:0] mappings of the PHY BIST Status signals, for automated test equipment sensing and debug.

**Table 20-2. PHY BIST Status at Device Ports**

Device Ball Name	PHY BIST Status
GPIO0	PHY BIST Error Status
GPIO1	PHY BIST Done Status
GPIO2	PHY BIST All Sync OK Status
GPIO3	PHY SerDes Quad 0 BIST Error Status
GPIO4	PHY SerDes Quad 1 BIST Error Status
GPIO5	PHY SerDes Quad 2 BIST Error Status
GPIO6	PHY SerDes Quad 3 BIST Error Status

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.5 Using the Even/Odd Port SerDes Quad x Diagnostic Data Registers

Each SerDes quad has its own Diagnostic Data register. The **Even/Odd Port SerDes Quad x Diagnostic Data** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **240h** through **24Ch**) contents reflect the performance of the SerDes selected by the registers' *SerDes Diagnostic Data Select* field [25:24], as defined in **Table 20-3**. This control is specific to this register (which reports results of UTP and PRBS 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. 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 20-3. Even/Odd Port SerDes Quad x Diagnostic Data Register Contents  
(Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port,  
offsets 240h through 24Ch)**

Register Offset	Register	<i>SerDes Diagnostic Data Select Field [25:24] Value</i>			
		00b	01b	10b	11b
240h	<b>Even Ports – SerDes Quad 0 Diagnostic Data</b>	SerDes 0	SerDes 1	SerDes 2	SerDes 3
248h	<b>Even Ports – SerDes Quad 1 Diagnostic Data</b>	SerDes 4	SerDes 5	SerDes 6	SerDes 7
244h	<b>Odd Ports – SerDes Quad 2 Diagnostic Data</b>	SerDes 8	SerDes 9	SerDes 10	SerDes 11
24Ch	<b>Odd Ports – SerDes Quad 3 Diagnostic Data</b>	SerDes 12	SerDes 13	SerDes 14	SerDes 15

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.6 Pseudo-Random and Bit-Pattern Generation

Each SerDes quad has an associated PRBS generator and checker. The PRBS generator is based upon a 7-bit **Linear Feedback Shift register (LFSR)**, which can generate up to  $(2^7 - 1)$  unique patterns. The PRBS logic is assigned to a SerDes in the quad, by manipulating the appropriate **Even/Odd Port SerDes Quad x Diagnostic Data** register *SerDes Diagnostic Data Select* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 240h through 24Ch, field [25:24]). The PRBS bit stream is used for Analog Far-End Loopback testing.

The PEX 8619 also provides a method of creating a repeating programmable bit pattern. Each of the four 32-bit **Physical Layer User Test Pattern, Bytes x through y** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 210h through 21Ch) are loaded with a 32-bit data pattern. After a Port is established as a Loopback Master, Set the appropriate **Even/Odd Port User Test Pattern Delay** register *SerDes x User Test Pattern Enable* bit(s) (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[23:16], and Odd Ports, offset 25Ch[23:16]), for the SerDes associated with that Port. The PEX 8619 proceeds to transmit the data pattern on all Lanes, starting with Byte 0 of the **Physical Layer User Test Pattern, Bytes 0 through 3** register and continuing, in sequence, through Byte 3 of the **Physical Layer User Test Pattern, Bytes 12 through 15** register. SKIP Ordered-Sets are inserted at the proper intervals, which makes this method appropriate for Digital Far-End Loopback testing. The received pattern is compared to the transmitted pattern. Any errors are logged and can be retrieved, by reading the appropriate **Even/Odd Port SerDes Quad x Diagnostic Data** register RO bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets 240h through 24Ch, bits [30 and 23:0]).

*To produce a pseudo-clock bitstream in Analog Loopback mode, Set the registers as follows:*

1. In the Slave device, enable Analog Loopback by Setting the **Common Physical Layer Safety and Test** register *Analog Loopback Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset 23Ch[21]).
2. In the PEX 8619 Loopback Master device, Set the Port's **Even/Odd Port Physical Layer Command/Status** register *Port x Loopback Command* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 220h[28, 24, 20, 16, 12, 8, 4, 0], and Odd Ports, offset 224h[28, 24, 20, 16, 12, 8, 4, 0]).
3. Check whether loopback is successful, by reading the Port's **Even/Odd Port Physical Layer Command/Status** register *Port x Ready as Loopback Master* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 220h[31, 27, 23, 19, 15, 11, 7, 3], and Odd Ports, offset 224h[31, 27, 23, 19, 15, 11, 7, 3]) in the same Nibble that was Set in step 2. The Nibble value is 9h if Loopback was successful.
4. Set the **Even/Odd Port User Test Pattern Delay** register *SerDes x PRBS Enable* bit (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset 258h[31:24], and Odd Ports, offset 25Ch[31:24]) for the SerDes associated with the Port selected in step 2.
5. Check the **Even/Odd Port Loopback Master Status/Control** register *PRBS Pattern Sync Status Device Lane x* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port – Even Ports, offset B80h[23:16], and Odd Ports, offset B84h[23:16]). A bit returning a value of 1 indicates that the looped-back PRBS pattern is detected at the Master's Receiver.
6. Exit Loopback mode, by Clearing the following bits:
  - a. *SerDes x PRBS Enable* bit, for the SerDes (selected in step 4) used by the Port selected in step 2.
  - b. *Port x Loopback Command* bit, for the Port selected in step 2.

Clearing these bits causes both sides of the Link to exit their LTSSM *Loopback* states, and return to the L0 Link PM state.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.7 PHY Testability Features

The PEX 8619 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 20-4 describes the Configuration bits.

**Table 20-4. Configuration Bits to Ease PHY Testability**

Register Bit(s) <sup>a</sup>	Description
<i>SerDes x Mask Electrical Idle Detect</i> <b>Even/Odd Port</b> <b>Electrical Idle for Compliance Mask</b> registers (Even Ports, offset 200h[15:8], and Odd Ports, offset 204h[15:8])	When any one of these bits is Set, the Electrical Idle condition flag of the corresponding Lane does not assert, regardless of the actual presence of Electrical Idle.
<i>SerDes x Mask Receiver Not Detected</i> <b>Even/Odd Port</b> <b>Receiver Detect Mask</b> registers (Even Ports, offset 200h[23:16], and Odd Ports, offset 204h[23:16])	When any one of these bits is Set, the PHY functions as if a Receiver was detected on the corresponding Lane, regardless of the actual presence of a Receiver.
<i>Test Pattern x</i> <b>Physical Layer User Test Pattern,</b> <b>Bytes x through y</b> registers (offsets 210h through 21Ch)	A 16-byte test pattern can be written to these four registers. When UTP transmission is enabled, Byte 0 of register offset 210h is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 21Ch is transmitted last. (Refer to <a href="#">Section 20.2.3</a> 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>Even/Odd Port</b> <b>Physical Layer Command/Status</b> registers (Even Ports, offset 220h[29, 25, 21, 17, 13, 9, 5, 1], and Odd Ports, offset 224h[29, 25, 21, 17, 13, 9, 5, 1])	Unconditionally disables the data scramblers on the Lanes of the corresponding Port, and causes the <i>Scrambler Disable</i> Training Control Bit to be Set in transmitted Training Sets. There is one bit for each Port.
<i>Disable Port x</i> <b>Even/Odd Port</b> <b>Disable</b> registers (Even Ports, offset 230h[7:0], and Odd Ports, offset 234h[7:0])	When Set, LTSSM remains in the <i>Detect.Quiet</i> state on Port 0, if it is currently in, or returns to, that state. Unconditionally disables the corresponding 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> state 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 the 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>Even/Odd Port</b> <b>Quiet</b> registers (Even Ports, offset 230h[15:8], and Odd Ports, offset 234h[15:8])	When Set, the LTSSM remains in the <i>Detect.Quiet</i> state if it is currently in, or returns to, that state. Unlike the <i>Disable Port x</i> bits, these bits do not force the LTSSM into the <i>Detect.Quiet</i> state. Once in the <i>Detect.Quiet</i> state, 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> state.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 20-4. Configuration Bits to Ease PHY Testability (Cont.)**

Register Bit(s) <sup>a</sup>	Description
<i>Port x Test Pattern x Rate</i> <b>Even/Odd Port</b> <b>Test Pattern Rate</b> registers (Even Ports, offset 230h[23:16], and Odd Ports, offset 234h[23:16])	The corresponding Port transmits the selected test pattern (PRBS or UTP) at 5.0 GT/s, if the Port's <i>Hold Port x Quiet</i> bit is also Set (manual rate selection is enabled only when the <i>Hold Port x Quiet</i> bit is Set).
<i>Port x Receiver Error Counter</i> <b>Even/Odd Port</b> <b>Receiver Error Counters</b> registers (offsets <b>B88h</b> through <b>B94h</b> )	Contains four 8-bit fields that, when read, return the quantity of Receiver errors detected by the corresponding Port. The Error Counter saturates at 255. The Counter is Cleared with any Write to the corresponding byte in these registers; otherwise, this register is RO.

- a. All registers listed in this table are located in Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.8 JTAG Interface

The PEX 8619 provides a Joint Test Action Group (JTAG) Boundary Scan interface, which is used to debug board connectivity for each ball.

### 20.8.1 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 20 MHz
- **JTAG Reset Requirements** – Refer to [Section 20.8.4](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.8.2 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 20-5 lists the JTAG instructions, along with their input codes; Instruction Length is 29 bits.

The PEX 8619 returns the JTAG IDCODE values listed in Table 20-6.

**Table 20-5. JTAG Instructions**

Instruction	Input Code	Comments
BYPASS	1FFF_FFFFh	<i>IEEE Standard 1149.1-1990</i>
EXTEST	1FFF_FFE8h	
SAMPLE	1FFF_FFF8h	
PRELOAD	1FFF_FFF8h	
EXTEST_PULSE	1FF7_FFE8h	<i>IEEE Standard 1149.6-2003</i>
EXTEST_TRAIN	1FD3_FFE8h	
CLAMP	1FFF_FFEFh	<i>IEEE Standard 1149.1-1990</i>
IDCODE	1FFF_FFFEH	

**Table 20-6. JTAG IDCODE Values**

Units	Version	Part Number	PLX Manufacturer Identity	Least Significant Bit
Bits	0000b	1000_0110_0001_1000b	001_1100_1101b	1
Hex	0h	8618h	1CDh	1h

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 20.8.3 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 8619.

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

**Note:** The PEX 8619 requires Fundamental Reset (**PEX\_PERST#** input assertion and de-assertion) and active REFCLK to initialize the switch, prior to using the JTAG interface.

## 20.8.4 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 8619's TAP Controller into the *Test-Logic-Reset* state, which selects the PEX 8619 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 8619 power-up, along with **PEX\_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 PEX 8619's JTAG TAP Controller is not intended to be used by the design, an external pull-down resistor is recommended on the JTAG\_TRST# input, because its internal pull-down resistor is weak.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 20.9 Lane Good Status LEDs

The PEX 8619 provides Lane Good outputs, `PEX_LANE_GOOD[15:0]#`, that can directly drive external common anode LED modules, to provide visual indication that the PHY of that Lane's Link is trained to at least x1 width.

Software can determine:

- Which Lanes have completed PHY linkup, by performing a Memory Read of the **Software Lane Status** register *Lane x Up Status* bits (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset `1F4h[15:0]`, which correspond to Lanes [15-0], respectively).
- Whether the Link for each Port has trained, by reading either the Port's **Link Status** register *Data Link Layer Link Active* bit (offset `78h[29]`), or Port's **VC0 Resource Status** register *VC0 Negotiation Pending* and **VC1 Resource Status** register *VC1 Negotiation Pending* bits (offsets `160h[17]` and `16Ch[17]`, respectively). If the *Data Link Layer Link Active* bit is Set, or *VC0 Negotiation Pending* and/or *VC1 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** and **VC1 Resource Status** registers 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.

**Table 20-7** describes the relationship of the LED On/Off patterns, as they relate to the Lane status indicated by `PEX_LANE_GOOD[15:0]#`.

**Table 20-7. PEX\_LANE\_GOOD[15:0]# LED On/Off Patterns, by State**

State	LED Pattern
Lane is disabled	Off
Lane is enabled, 5.0 GT/s	On
Lane is enabled, 2.5 GT/s, reduced Lanes are up	Blinking, 0.5 seconds On, 0.5 seconds Off

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Chapter 21 Electrical Specifications

### 21.1 Introduction

This chapter provides the PEX 8619 electrical specifications.

### 21.2 Power-Up/Power-Down Sequence

The PEX 8619 does not have power sequencing requirements. The power rails can be powered up and powered down, in any sequence.

### 21.3 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 8619 at these limits is not recommended.*

**Table 21-1. Absolute Maximum Rating (All Voltages Referenced to VSS System Ground)**

Item	Symbol	Absolute Maximum Rating	Units
I/O Interface Supply Voltage	VDD25	-0.5 to +3.6	V
PLL Supply Voltage	VDD25A	-0.5 to +3.6	V
Core (Logic) Supply Voltage	VDD10	-0.3 to +1.5	V
SerDes Analog Supply Voltage	VDD10A	-0.3 to +1.5	V
Input Voltage (2.5V Interface)	V <sub>I</sub>	-0.3 to +3.6	V
Operating Ambient Temperature Commercial Industrial	T <sub>A</sub>	0 to +70 -40 to +85	°C
Storage Temperature	T <sub>STG</sub>	-65 to +150	°C

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 21.4 Power Characteristics

**Table 21-2. Operating Condition Power Supply Rails**

Symbol	Parameter	Min	Typ	Max	Units
VDD10	Digital Core Supply {1.0V $\pm$ 5%}	0.95	1.0	1.05	V
VDD10A	Analog SerDes Supply {1.0V $\pm$ 5%}	0.95	1.0	1.05	V
VDD25	I/O Supply {2.5V $\pm$ 10%}	2.25	2.50	2.75	V
VDD25A	Phase-Locked Loop (PLL) Supply {2.5V $\pm$ 10%}	2.25	2.50	2.75	V

## 21.5 Power Consumption Estimates

Table 21-3 and Table 21-4 list the PEX 8619 power consumption estimates at Gen 1 and Gen 2 Link speeds, respectively.

**Table 21-3. PEX 8619 Gen 1 (2.5 GT/s) Power Consumption Estimates (Watts)**

Lanes	Ports	Core Digital (VDD10) (Amps)		SerDes Analog (VDD10A) (Amps)		SerDes Digital (VDD10) (Amps)		PLL and I/O (VDD25A/VDD25) (Amps)		Total (Watts)	
		Typ	Max	Typ	Max	Typ	Max	Typ	Max	Typ <sup>a</sup>	Max <sup>b, c</sup>
16	16	0.99	1.86	0.47	0.86	0.10	0.22	0.05	0.05	1.70	3.23

- a. Typical power based upon 35% traffic, all Lanes in active L0 Link PM state, typical power rails (1.0V/2.5V).
- b. Maximum power based upon 85% traffic, idle Lanes in active L0s Link PM state, maximum power rails (1.05V/2.75V).
- c. Maximum power is at high temperature and Fast/Fast (FF) process corner silicon.

**Table 21-4. PEX 8619 Gen 2 (5.0 GT/s) Power Consumption Estimates (Watts)**

Lanes	Ports	Core Digital (VDD10) (Amps)		SerDes Analog (VDD10A) (Amps)		SerDes Digital (VDD10) (Amps)		PLL and I/O (VDD25A/VDD25) (Amps)		Total (Watts)	
		Typ	Max	Typ	Max	Typ	Max	Typ	Max	Typ <sup>a</sup>	Max <sup>b, c</sup>
16	16	1.21	2.12	0.53	1.01	0.12	0.27	0.05	0.05	1.99	3.70

- a. Typical power based upon 35% traffic, all Lanes in active L0 Link PM state, typical power rails (1.0V/2.5V).
- b. Maximum power based upon 85% traffic, idle Lanes in active L0s Link PM state, maximum power rails (1.05V/2.75V).
- c. Maximum power is at high temperature and Fast/Fast (FF) process corner silicon.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 21.6 I/O Interface Signal Groupings

**Table 21-5. Signal Group PCI Express Analog Interface**

Signal Group	Signal Type	Signals	Notes
(a)	PCI Express Output (Transmit)	PEX_PETn[15:0], PEX_PETp[15:0]	Refer to <a href="#">Table 21-7</a> and <a href="#">Table 21-8</a>
(b)	PCI Express Input (Receive)	PEX_PERn[15:0], PEX_PERp[15:0]	Refer to <a href="#">Table 21-7</a> and <a href="#">Table 21-9</a>
(c)	PCI Express Differential Clock Input	PEX_REFCLKn, PEX_REFCLKp, PEX_REFCLK_CFCn, PEX_REFCLK_CFCp	Refer to <a href="#">Table 21-7</a> and <a href="#">Table 21-10</a>
(d)	SerDes External Resistor	REXT_A[3:0], REXT_B[3:0]	$1.43K\Omega \pm 1\%$ , and refer to <a href="#">Table 21-7</a>

**Table 21-6. Signal Group Digital Interface**

Signal Group	Signal Type	Signals	Note
(e1)	Digital Input <sup>a</sup>	JTAG_TCK, JTAG_TDI, JTAG_TMS, JTAG_TRST#, STRAP_DEBUG_SEL0, STRAP_FAST_BRINGUP#, STRAP_NT_ENABLE#, STRAP_NT_UPSTRM_PORTSEL[3:0], STRAP_PLL_BYPASS#, STRAP_PORTCFG[3:0], STRAP_PROBE_MODE#, STRAP_RESERVED16, STRAP_RESERVED17#, STRAP_SERDES_MODE_EN#, STRAP_SSC_ISO_ENABLE#, STRAP_TESTMODE[3:0], STRAP_UPSTRM_PORTSEL[3:0]	
(e2)	Digital Input <sup>a</sup>	I2C_ADDR[2:0]	
(f)	Digital Input with Internal Pull-up Resistor	PEX_PERST#, SHPC_INT#, STRAP_NT_P2P_EN#, STRAP_SMBUS_EN#, STRAP_UPCFG_TIMER_EN#	Refer to <a href="#">Table 21-7</a>
(g)	Digital Tri-State Output (8 mA)	EE_DI, FATAL_ERR#, JTAG_TDO, PEX_NT_RESET#, STRAP_SPARE0#	
(h)	Bidirectional with Internal Pull-up Resistor (8 mA Drive)	EE_CS#, EE_DO, EE_SK, GPIO[31:0], PEX_LANE_GOOD[15:0]#, STRAP_SPARE[5, 1]#	
(i)	Bidirectional (Open Drain) with Internal Pull-up Resistor	PEX_INTA#	
(j)	Bidirectional (Open Drain) Schmitt Trigger Input	I2C_SCL0, I2C_SCL1, I2C_SDA0, I2C_SDA1	

a. These signals must be pulled or tied High to VDD25 or Low to VSS (GND), per the instructions provided in [Section 3.4, “Signal Ball Descriptions.”](#)

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-7. Analog and Digital Interfaces (All Signal Groups) – DC Electrical Characteristics**

<b>Symbol</b>	<b>Signal Group</b>	<b>Parameter</b>	<b>Min</b>	<b>Typ</b>	<b>Max</b>	<b>Unit</b>	<b>Conditions</b>
I <sub>OL</sub>	(g) (h) (i) (j)	Output Low Current	13	22	32	mA	V <sub>OLmax</sub> = 0.7V
I <sub>OH</sub>	(g) (h)	Output High Current	8	16	27	mA	V <sub>OHmin</sub> = 1.7V
V <sub>IL</sub>	(e1) (e2) (f) (h) (i) (j)	Input Low Voltage	-0.3		0.7	V	
V <sub>IH</sub>	(e1) (e2) (f) (h) (i) (j)	Input High Voltage	1.7		2.8	V	Refer to Note 1.
V <sub>T</sub>	(e1) (e2) (h) (i)	Threshold Point	0.97	1.05	1.14	V	
C <sub>PIN</sub>	(a) (b) (c) (d) (e1) (e2) (f) (g) (h) (i) (j)	Ball Capacitance			5	pF	
I <sub>LEAKAGE</sub>	(g)	Tri-State Leakage			±10	µA	
	(e1) (f)	Input Leakage			±10	µA	
	(e2)	Input Leakage	-22.6		-47.5	µA	
	(f) (h) (i)	Pull-Up Leakage	-22.6		-47.5	µA	
R <sub>PU</sub>	(f) (h) (i)	Pull-Up Impedance	74K	111K	178K	Ω	
V <sub>T</sub>	(j)	Schmitt Trigger Rising Threshold	1.2	1.3	1.4	V	
		Schmitt Trigger Falling Threshold	0.84	0.93	1.01	V	
V <sub>HYS</sub>	(j)	Input Hysteresis	360	370	390	mV	

**Note:**

1. The specified maximum V<sub>IH</sub> is for recommended operating conditions. Because these I/O buffers are 3.3V tolerant, a maximum V<sub>IH</sub> of 3.6V can safely be applied to these signal balls.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-8. 2.5 and 5.0 GT/s PCI Express Transmitter (Signal Group a) – AC and DC Characteristics**

Symbol	Parameter	2.5 GT/s	5.0 GT/s	Units	Comments
UI	Unit Interval	399.88 (min) 400.12 (max)	199.94 (min) 200.06 (max)	ps	The specified UI is equivalent to a tolerance of $\pm 300$ ppm. UI does not account for variations caused by Spread-Spectrum Clock (SSC). Refer to Note 1.
$V_{TX-DIFF-PP}$	Differential Peak-to-Peak Output Voltage	0.8 (min) 1.2 (max)	0.8 (min) 1.2 (max)	V	Measured with compliance test load. $V_{TX-DIFF-PP} = 2 \times  V_{TX-D+} - V_{TX-D-} $
$V_{TX-DIFF-PP-LOW}$	Low Power Differential Peak-to-Peak Output Voltage	0.4 (min) 1.2 (max)	0.4 (min) 1.2 (max)	V	Measured with compliance test load. $V_{TX-DIFF-PP-LOW} = 2 \times  V_{TX-D+} - V_{TX-D-} $ Must be implemented with no de-emphasis.
$V_{TX-DE-RATIO-3.5dB}$	Tx De-Emphasis Level Ratio	3.0 (min) 4.0 (max)	3.0 (min) 4.0 (max)	dB	Ratio of the $V_{TX-DIFF-PP}$ of the 2 <sup>nd</sup> and following bits after a transition, divided by the $V_{TX-DIFF-PP}$ of the 1 <sup>st</sup> bit after a transition. Refer to Note 2.
$V_{TX-DE-RATIO-6dB}$	Tx De-Emphasis Level Ratio	N/A	5.5 (min) 6.5 (max)	dB	Ratio of the $V_{TX-DIFF-PP}$ of the 2 <sup>nd</sup> and following bits after a transition, divided by the $V_{TX-DIFF-PP}$ of the 1 <sup>st</sup> bit after a transition. Refer to Note 2.
$T_{MIN-PULSE}$	Instantaneous Pulse Width (including all jitter sources)	Not specified	0.9 (min)	UI	Measured relative to rising/falling pulse. Refer to Note 3.
$T_{TX-EYE}$	Minimum Tx Eye Width	0.75 (min)	0.75 (min)	UI	Does not include SSC nor REFCLK jitter. Includes $R_j$ at $10^{-12}$ . Refer to Notes 3 and 4.
$T_{TX-EYE-MEDIAN-to-MAX-JITTER}$	Maximum Time between the Jitter Median and Maximum Deviation from the Median	0.125 (max)	Not specified	UI	Measured differentially at zero crossing points, after applying the 2.5 GT/s Clock Recovery function. Refer to Note 3.
$T_{TX-HF-DJ-DD}$	Tx Deterministic Jitter > 1.5 MHz	Not specified	0.15 (max)	UI	Deterministic jitter only. Refer to Note 3.
$T_{TX-LF-RMS}$	Tx RMS Jitter < 1.5 MHz	Not specified	3.0	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)	UI	Measured differentially from 20 to 80% of swing. Refer to Note 3.
$T_{RF-MISMATCH}$	Tx Rise/Fall Mismatch	Not specified	0.1 (max)	UI	Measured from 20 to 80% differentially. Refer to Note 3.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-8. 2.5 and 5.0 GT/s PCI Express Transmitter (Signal Group a) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	2.5 GT/s	5.0 GT/s	Units	Comments
BW <sub>TX-PLL</sub>	Maximum Tx PLL Bandwidth	22 (max)	16 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 5.
BW <sub>TX-PLL-LO-3DB</sub>	Minimum Tx PLL Bandwidth for 3-dB Peaking	1.5 (min)	8 (min)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Notes 5 and 7.
BW <sub>TX-PLL-LO-1DB</sub>	Minimum Tx PLL Bandwidth for 1-dB Peaking	Not specified	5 (min)	MHz	
PKG <sub>TX-PLL1</sub>	TX PLL Peaking with 8-MHz Minimum Bandwidth	Not specified	3.0 (max)	dB	
PKG <sub>TX-PLL2</sub>	TX PLL peaking with 5-MHz Minimum Bandwidth	Not specified	1.0 (max)	dB	Refer to Note 7.
RL <sub>TX-DIFF</sub>	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	dB	
RL <sub>TX-CM</sub>	TX Common Mode Return Loss (Package + Silicon)	6 (min)	6 (min)	dB	S <sub>11</sub> parameter. 2.5 GT/s – Measured over 0.05- to 1.25-GHz range. 5.0 GT/s – Measured over 0.05- to 2.5-GHz range.
Z <sub>TX-DIFF-DC</sub>	DC Differential Tx Impedance	80 (min) 120 (max)	120 (max)	Ω	Tx DC Differential mode low impedance. Parameter is captured for 5.0 GHz by RL <sub>TX-DIFF</sub> .
V <sub>TX-CM-AC-PP</sub>	Tx AC Common Mode Voltage (5.0 GT/s)	Not specified	100 (max)	mVPP	Refer to Note 6.
V <sub>TX-CM-AC-P</sub>	Tx AC Common Mode Voltage (2.5 GT/s)	20 (max)	Not specified	mVPP	Refer to Note 6.
I <sub>TX-SHORT</sub>	Tx Short Circuit Current Limit	90 (max)	90 (max)	mA	Total current the Transmitter can provide when shorted to its ground.
V <sub>TX-DC-CM</sub>	Tx DC Common Mode Voltage	0 (min) 3.6 (max)	0 (min) 3.6 (max)	V	Allowed DC common mode voltage, under any conditions.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

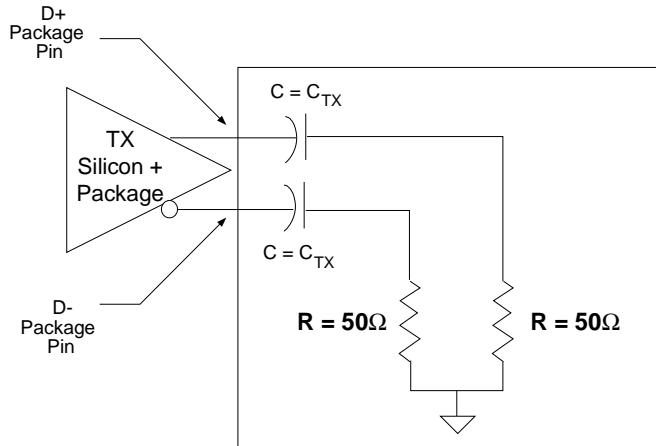
**Table 21-8. 2.5 and 5.0 GT/s PCI Express Transmitter (Signal Group a) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	2.5 GT/s	5.0 GT/s	Units	Comments
$V_{TX-CM-DC-ACTIVE-IDLE-DELTA}$	Absolute Delta of DC Common Mode Voltage during L0 Link PM state and Electrical Idle	0 (min) 100 (max)	0 (min) 100 (max)	mV	$ V_{TX-CM-DC} \text{ [during L0]} - V_{TX-CM-Idle-DC} \text{ [during Electrical Idle]}  \leq 100 \text{ mV}$ $V_{TX-CM-DC} = \text{DC}_{(\text{avg})} \text{ of }  V_{TX-D+} + V_{TX-D-}  / 2 \text{ [L0]}$ $V_{TX-CM-Idle-DC} = \text{DC}_{(\text{avg})} \text{ of }  V_{TX-D+} + V_{TX-D-}  / 2 \text{ [Electrical Idle]}$
$V_{TX-CM-DC-LINE-DELTA}$	Absolute Delta of DC Common Mode Voltage between D+ and D-	0 (min) 25 (max)	0 (min) 25 (max)	mV	$ V_{TX-CM-DC-D+} - V_{TX-CM-DC-D-}  \leq 25 \text{ mV}$ $V_{TX-CM-DC-D+} = \text{DC}_{(\text{avg})} \text{ of }  V_{TX-D+} $ $V_{TX-CM-DC-D-} = \text{DC}_{(\text{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)	mV	$V_{TX-IDLE-DIFF-p} =  V_{TX-IDLE-D+} - V_{TX-IDLE-D-}  \leq 20 \text{ mV}$ Voltage must be high-pass filtered, to remove any DC component.
$V_{TX-IDLE-DIFF-DC}$	DC Electrical Idle Differential Peak Output Voltage	Not specified	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.
$V_{TX-RCV-DETECT}$	Amount of Voltage Change Allowed during Receiver Detection	600 (max)	600 (max)	mV	Total amount of voltage change that a Transmitter can apply, to sense whether a low-impedance Receiver is present.
$T_{TX-IDLE-MIN}$	Minimum Time Spent in Electrical Idle	20 (min)	20 (min)	ns	Minimum time a Transmitter must be in Electrical Idle. Used by the Receiver to start looking for an Electrical Idle Exit after successfully receiving an Electrical Idle Ordered-Set (EIOS).
$T_{TX-IDLE-SET-TO-IDLE}$	Maximum Time to Transition to a Valid Electrical Idle after Sending an Electrical Idle Ordered-Set	8 (max)	8 (max)	ns	After sending the required 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 Tx 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)	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)	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)	ps	Static skew between any two Lanes within a single Transmitter.
$C_{TX}$	AC-Coupling Capacitor	75 (min) 200 (max)	75 (min) 200 (max)	nF	All Transmitters shall be AC-coupled. The AC coupling is required either within the media, or within the transmitting component itself.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Notes:**

1. *SSC permits a +0, -5,000 ppm modulation of the clock frequency, at a modulation rate not to exceed 33 kHz.*
2. *Specified at the measurement point into a timing and voltage compliance test load, as illustrated in Figure 21-1.*

**Figure 21-1. Compliance Test/Measurement Load**

3. *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 a scope with at least 6.2 GHz bandwidth. Measurements at 5.0 GT/s must de-convolve effects of the 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 balls; however, de-convolution is recommended. At least  $10^6$  UI of data must be acquired.*
4. *Transmitter jitter is measured by driving the Tx under test with a low jitter “ideal” clock and connecting the device under test (DUT) to a reference load.*
5. *The Tx PLL bandwidth must lie between the minimum and maximum ranges listed in Table 21-8. PLL peaking must lie below the values listed in Table 21-8.*  
*The PLL bandwidth extends from zero (0) up to the value(s) specified in Table 21-8.*
6. *Measurement is made over at least  $10^6$  UI.*
7. *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.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

**Table 21-9. 2.5 and 5.0 GT/s PCI Express Receiver (Signal Group b) – AC and DC Characteristics**

<b>Symbol</b>	<b>Parameter</b>	<b>2.5 GT/s</b>	<b>5.0 GT/s</b>	<b>Units</b>	<b>Comments</b>
UI	Unit Interval	399.88 (min) 400.12 (max)	199.94 (min) 200.06 (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)	V	$V_{RX-DIFF-PP} = 2 \times  V_{RX-D+} - V_{RX-D-} $
$T_{RX-EYE}$	Receiver Eye Time Opening	0.40 (min)	N/A	UI	Minimum eye time at Rx pins 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	N/A	0.40 (max)	UI	Maximum Rx inherent total timing error for common REFCLK Rx architecture. Refer to Note 1.
$T_{RX-DJ-DD-CC}$	Maximum Rx Inherent Deterministic Timing Error	N/A	0.30 (max)	UI	Maximum Rx inherent deterministic timing error for common REFCLK Rx architecture. Refer to Note 1.
$T_{RX-EYE-MEDIAN-to-MAX-JITTER}$	Maximum Time Delta between the Median and Deviation from the Median	0.3 (max)	Not specified	UI	
$T_{RX-MIN-PULSE}$	Minimum Width Pulse at Rx	Not specified	0.6 (min)	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)	Ratio	Rx eye must simultaneously meet $V_{RX-EYE}$ limits.
$BW_{RX-PLL-HI}$	Maximum Rx PLL Bandwidth	22 (max)	16 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 2.
$BW_{RX-PLL-LO-3DB}$	Minimum Rx PLL Bandwidth for 3-dB Peaking	1.5 min	8 (min)	MHz	
$BW_{RX-PLL-LO-1DB}$	Minimum Rx PLL Bandwidth for 1-dB Peaking	Not specified	5 (min)	MHz	
PKG <sub>RX-PLL1</sub>	Rx PLL Peaking with 8-MHz Minimum Bandwidth	Not specified	3.0	dB	
PKG <sub>RX-PLL2</sub>	Rx PLL Peaking with 5-MHz Minimum Bandwidth	Not specified	1.0	dB	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-9. 2.5 and 5.0 GT/s PCI Express Receiver (Signal Group b) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	2.5 GT/s	5.0 GT/s	Units	Comments
RL <sub>RX-DIFF</sub>	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	dB	Refer to Note 3.
RL <sub>RX-CM</sub>	Common Mode Return Loss	6 (min)	6 (min)	dB	Refer to Note 3.
Z <sub>RX-DC</sub>	Rx DC Single-Ended Impedance	40 (min) 60 (max)	40 (min) 60 (max)	Ω	Required Rx D+ and D- DC impedance ( $50\Omega \pm 20\%$ tolerance). Refer to Note 4.
Z <sub>RX-DIFF-DC</sub>	DC Differential Rx Impedance	80 (min) 120 (max)	Not specified	Ω	Rx DC Differential mode impedance. Parameter is captured for 5.0 GHz by RL <sub>RX-DIFF</sub> . Refer to Note 4.
V <sub>RX-CM-AC-P</sub>	Rx AC Common Mode Voltage	150 (max)	150 (max)	mVP	Measured at Rx pins, into a pair of $50\Omega$ terminations into Ground. Refer to Note 5.
Z <sub>RX-HIGH-IMP-DC-POS</sub>	DC Input Common Mode Input Impedance for Voltage >0 during Reset or Power-Down	50K (min)	50K (min)	Ω	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 <sub>RX-HIGH-IMP-DC-NEG</sub>	DC Input Common Mode Input Impedance for Voltage <0 during Reset or Power-Down	1.0K (min)	1.0K (min)	Ω	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.
V <sub>RX-IDLE-DET-DIFFp-p</sub>	Electrical Idle Detect Threshold	65 (min) 175 (max)	65 (min) 175 (max)	mV	$V_{RX-IDLE-DET-DIFFp-p} = 2 \times  V_{RX-D+} - V_{RX-D-} $ Measured at the Receiver's package pins.
T <sub>RX-IDLE-DET-DIFF-ENTERTIME</sub>	Unexpected Electrical Idle Enter Idle Detect Threshold Integration Time	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 <sub>RX-IDLE-DET-DIFF-ENTERTIME</sub> to signal an unexpected idle condition.
L <sub>RX-SKEW</sub>	Total Lane-to-Lane Skew	20 (max)	8 (max)	ns	Across all Lanes on a Port. Includes variation in the length of a SKIP Ordered-Set at the Rx, as well as any delay differences arising from the interconnect itself. Refer to Note 7.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**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 21-9](#). 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 occurs properly. 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:

$$\max\{ |(Vd+ - Vd-) - V_{CMDC}| \}$$

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. The  $L_{RX-SKEW}$  parameter exists to handle repeaters that re-generate REFCLK and introduce differing numbers of skips on different Lanes.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

**Table 21-10. PCI Express Differential Clock (Signal Group c) – AC and DC Characteristics**

Symbol	Parameter	Min	Typ	Max	Unit	Notes
F <sub>REFCLK</sub>	Reference Clock Frequency		100		MHz	<a href="#">1</a>
T <sub>REFCLK-HF-RMS</sub>	High frequency jitter -> 1.5 MHz to Nyquist RMS jitter after applying filter functions, per the <i>PCI Express Base r2.0</i>			3.1	ps RMS	
T <sub>REFCLK-LF-RMS</sub>	Low frequency jitter – 10 kHz to 1.5 MHz jitter after applying filter functions, per the <i>PCI Express Base r2.0</i>			3.0	ps RMS	
T <sub>REFCLK-SSC-RES</sub>	SSC residual after applying filter functions, per the <i>PCI Express Base r2.0</i>			75	ps	
V <sub>SW</sub>	Differential Voltage Swing (0-to-peak)	125	200	800	mV	
	Differential Voltage Swing (peak-to-peak)	250	400	1,600	mV	
T <sub>R/T<sub>F</sub></sub>	Clock Input Rise/Fall Time	0.6		4.0	V/ns	<a href="#">2</a>
DC <sub>REFCLK</sub>	Input Clock Duty Cycle	45	50	55	%	
R <sub>TERM</sub>	Input Parallel Termination (Single-ended)		50		Ω	
	Input Parallel Termination (Differential)		100		Ω	
PPM	Reference Clock Tolerance	-300		+300	ppm	

**Notes:**

1. *PEX\_REFCLKn/p* must be AC-coupled. Use a 0.01 to 0.1  $\mu$ F capacitor.
2. Specified at 20 to 80% points at the package balls.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 21.7 Transmit Drive Characteristics

The Drive Current and Transmit Equalization functions are programmable, to allow for optimization of different backplane lengths and materials.

The Transmit Drive Level is programmable (5-bit, per SerDes/Lane), to provide differential swing within the range listed in [Table 21-11](#). The **SerDes Drive Level x** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **B98h**, **B9Ch**, **BA0h**, and **BA4h**) provide access to all 16 Lanes for Drive Level programmability.

The Transmitter also incorporates programmable (5-bit, per SerDes/Lane) de-emphasis, to provide equalization to compensate for FR4 channel effects within the range listed in [Table 21-11](#). The **Post-Cursor Emphasis Level x** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **BA8h**, **BACh**, **BB0h**, and **BB4h**) provide access to all 16 Lanes for de-emphasis programmability.

The `Transmit_Drive_Level[4:0]` and `Post-Cursor_EmpHASis_Level[4:0]` bits are used together to program the differential swing, as well as the dB loss for optimum Tx drive across the intended backplane.

[Table 21-11](#) lists all possible combinations of Tx `DRV_LVL[4:0]` and `POST_CURSOR[4:0]`, to achieve minimum 800 mV transition amplitude and the resulting de-emphasis (in decibels, dB). Of these, only certain combinations yield the specified 3 to 4 dB or 5.5 to 6.5 dB de-emphasis, per the *PCI Express Base r2.0* (**highlighted in bold**). All combinations are listed, however, to provide maximum flexibility for fine-tuning the Tx drive characteristics to a specific backplane.

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels**

POST_CURSOR [4:0]	DRV_LVL [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	De-Emphasis (dB)	Notes
01h	11h	820	789	0.34	
	12h	849	818	0.33	
	13h	876	845	0.31	
02h	10h	799	742	0.65	
	11h	830	773	0.61	
	12h	858	802	0.58	
	13h	884	829	0.56	
03h	10h	809	727	0.93	
	11h	839	758	0.88	
	12h	867	787	0.84	
	13h	893	814	0.80	
04h	10h	818	712	1.22	
	11h	848	743	1.15	
	12h	876	772	1.09	
	13h	901	799	1.04	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
05h	0Fh	797	664	1.59	
	10h	828	697	1.50	
	11h	857	728	1.41	
	12h	884	758	1.34	
	13h	909	785	1.27	
	1Fh	796	663	1.59	
06h	0Fh	806	649	1.88	
	10h	837	682	1.77	
	11h	866	714	1.68	
	12h	892	743	1.59	
	13h	916	770	1.51	
	1Fh	806	649	1.88	
07h	0Fh	816	635	2.18	
	10h	846	668	2.05	
	11h	874	700	1.94	
	12h	900	729	1.83	
	13h	924	756	1.74	
	1Fh	816	635	2.18	
08h	0Fh	825	620	2.48	
	10h	855	654	2.33	
	11h	883	685	2.20	
	12h	908	715	2.08	
	13h	931	742	1.98	
	1Fh	825	620	2.47	
09h	0Eh	802	571	2.95	
	0Fh	834	607	2.77	
	10h	863	640	2.60	
	11h	891	671	2.46	
	12h	916	701	2.33	
	13h	938	728	2.21	
	1Eh	802	571	2.96	
	1Fh	834	606	2.77	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

POST_CURSOR [4:0]	DRV_LVL [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	De-Emphasis (dB)	Notes
0Ah	0Eh	811	557	3.27	
	0Fh	843	593	3.06	
	10h	872	626	2.88	
	11h	899	658	2.71	
	12h	923	687	2.57	
	13h	945	714	2.43	
	1Eh	811	557	<b>3.27</b>	
	1Fh	843	593	<b>3.06</b>	
0Bh	0Eh	821	543	<b>3.58</b>	
	0Fh	851	579	<b>3.35</b>	
	10h	880	612	<b>3.15</b>	
	11h	906	644	2.97	
	12h	930	673	2.81	
	13h	951	700	2.66	
	1Eh	820	543	<b>3.58</b>	
	1Fh	851	579	<b>3.35</b>	
0Ch	0Dh	797	492	4.19	
	0Eh	829	530	<b>3.89</b>	
	0Fh	860	566	<b>3.64</b>	
	10h	888	599	<b>3.42</b>	
	11h	914	630	<b>3.22</b>	
	12h	937	660	<b>3.05</b>	
	13h	958	687	2.89	
	1Dh	797	492	4.19	
	1Eh	829	530	<b>3.89</b>	
	1Fh	860	565	<b>3.64</b>	
0Dh	0Dh	806	479	4.52	
	0Eh	838	517	4.20	
	0Fh	868	552	<b>3.93</b>	
	10h	896	586	<b>3.69</b>	-3.5 dB default for PEX 8619
	11h	921	617	<b>3.48</b>	
	12h	944	646	<b>3.29</b>	
	13h	964	673	<b>3.11</b>	
	1Dh	806	479	4.52	
	1Eh	838	516	4.20	
	1Fh	868	552	<b>3.93</b>	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
0Eh	0Dh	815	466	4.86	
	0Eh	847	503	4.51	
	0Fh	876	539	4.22	
	10h	903	572	<b>3.96</b>	
	11h	928	604	<b>3.73</b>	
	12h	950	633	<b>3.53</b>	
	13h	970	660	<b>3.34</b>	
	1Dh	815	466	4.86	
	1Eh	846	503	4.52	
	1Fh	876	539	4.22	
0Fh	0Dh	824	453	5.20	
	0Eh	855	490	4.83	
	0Fh	884	526	4.51	
	10h	911	559	4.23	
	11h	935	591	<b>3.98</b>	
	12h	957	620	<b>3.76</b>	
	13h	975	647	<b>3.56</b>	
	1Dh	823	453	5.20	
	1Eh	855	490	4.83	
	1Fh	884	526	4.51	
10h	0Ch	799	399	<b>6.02</b>	
	0Dh	832	439	<b>5.55</b>	
	0Eh	863	477	5.15	
	0Fh	892	513	4.81	
	10h	918	546	4.51	
	11h	942	578	4.25	
	12h	963	607	4.01	
	13h	981	634	3.79	
	1Ch	799	399	<b>6.02</b>	
	1Dh	832	439	<b>5.55</b>	
	1Eh	863	477	5.15	
	1Fh	892	513	4.81	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

POST_CURSOR [4:0]	DRV_LVL [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	De-Emphasis (dB)	Notes
11h	0Ch	808	387	<b>6.40</b>	
	0Dh	841	427	<b>5.89</b>	
	0Eh	871	464	5.46	
	0Fh	899	500	5.10	
	10h	925	533	4.78	
	11h	948	565	4.50	
	12h	969	594	4.24	
	13h	986	621	4.02	
	1Ch	808	387	<b>6.40</b>	
	1Dh	840	427	<b>5.89</b>	
	1Eh	871	464	5.46	
	1Fh	899	500	5.10	
12h	0Ch	816	374	6.77	
	0Dh	849	414	<b>6.23</b>	
	0Eh	879	452	<b>5.78</b>	
	0Fh	906	487	5.39	
	10h	932	521	5.05	
	11h	954	552	4.75	
	12h	974	582	4.48	
	13h	991	609	4.24	
	1Ch	816	374	6.78	
	1Dh	849	414	6.23	
	1Eh	879	452	5.78	
	1Fh	906	487	5.39	
13h	0Ch	825	362	7.16	
	0Dh	857	402	6.58	
	0Eh	886	439	6.09	
	0Fh	913	475	5.68	
	10h	938	509	5.32	
	11h	960	540	5.00	
	12h	980	569	4.72	
	13h	996	596	4.46	
	1Bh	795	326	7.76	
	1Ch	825	362	7.16	
	1Dh	856	402	6.58	
	1Eh	886	439	<b>6.09</b>	
	1Fh	913	475	<b>5.68</b>	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
14h	0Bh	799	308	8.29	
	0Ch	833	350	7.54	
	0Dh	864	389	6.93	
	0Eh	893	427	<b>6.41</b>	
	0Fh	920	463	<b>5.97</b>	
	10h	944	496	<b>5.59</b>	
	11h	966	528	5.25	
	12h	985	557	4.95	
	13h	1,001	584	4.68	
	1Bh	804	313	8.19	
	1Ch	833	349	7.54	
	1Dh	864	389	6.93	
	1Eh	893	427	<b>6.41</b>	
	1Fh	920	463	<b>5.97</b>	
15h	0Bh	808	296	8.73	
	0Ch	841	337	7.93	
	0Dh	872	377	7.28	
	0Eh	901	415	6.73	-6 dB default for PEX 8619
	0Fh	927	451	<b>6.26</b>	
	10h	950	484	<b>5.86</b>	
	11h	972	516	<b>5.50</b>	
	12h	990	545	5.19	
	13h	1,006	572	4.90	
	1Bh	812	301	8.62	
	1Ch	841	337	7.93	
	1Dh	872	377	7.28	
	1Eh	900	415	6.73	
	1Fh	927	451	<b>6.62</b>	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

POST_CURSOR [4:0]	DRV_LVL [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	De-Emphasis (dB)	Notes
16h	0Bh	816	284	9.18	
	0Ch	849	326	8.32	
	0Dh	879	365	7.63	
	0Eh	907	403	7.05	
	0Fh	933	439	6.56	
	10h	956	472	<b>6.13</b>	
	11h	977	504	<b>5.76</b>	
	12h	995	533	5.42	
	13h	1,010	560	5.12	
	1Bh	820	289	9.05	
	1Ch	849	325	8.33	
	1Dh	879	365	7.63	
	1Eh	907	403	7.05	
	1Fh	933	439	6.56	
17h	0Bh	824	272	9.63	
	0Ch	856	314	8.72	
	0Dh	886	354	7.98	
	0Eh	914	391	7.37	
	0Fh	939	427	6.85	
	10h	962	460	<b>6.40</b>	
	11h	982	492	<b>6.01</b>	
	12h	999	521	<b>5.65</b>	
	13h	1,014	548	5.34	
	1Bh	828	277	9.50	
	1Ch	856	314	8.72	
	1Dh	886	353	7.99	
	1Eh	914	391	7.37	
	1Fh	939	427	6.85	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
18h	0Ah	798	216	11.35	
	0Bh	832	260	10.10	
	0Ch	864	302	9.13	
	0Dh	893	342	8.35	
	0Eh	921	380	7.70	
	0Fh	945	415	7.15	
	10h	968	449	6.68	
	11h	987	480	<b>6.26</b>	
	12h	1,004	509	<b>5.89</b>	
	13h	1,018	537	<b>5.56</b>	
	1Ah	802	222	11.16	
	1Bh	836	266	9.96	
	1Ch	864	302	9.13	
	1Dh	893	342	8.35	
	1Eh	921	379	7.70	
	1Fh	945	415	7.15	
19h	0Ah	806	204	11.91	
	0Bh	840	249	10.57	
	0Ch	871	290	9.54	
	0Dh	900	330	8.71	
	0Eh	927	368	8.02	
	0Fh	951	404	7.45	
	10h	973	437	6.95	
	11h	992	469	6.51	
	12h	1,008	498	<b>6.12</b>	
	13h	1,021	525	<b>5.78</b>	
	1Ah	810	210	11.71	
	1Bh	844	254	10.42	
	1Ch	871	290	9.54	
	1Dh	900	330	8.71	
	1Eh	927	368	8.03	
	1Fh	951	404	7.45	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

POST_CURSOR [4:0]	DRV_LVL [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	De-Emphasis (dB)	Notes
1Ah	0Ah	814	193	12.49	
	0Bh	847	237	11.05	
	0Ch	878	279	9.95	
	0Dh	907	319	9.08	
	0Eh	933	357	8.35	
	0Fh	957	392	7.74	
	10h	978	426	7.22	
	11h	996	457	6.76	
	12h	1,012	487	<b>6.36</b>	
	13h	1,025	514	<b>6.00</b>	
	1Ah	818	199	12.28	
	1Bh	851	243	10.89	
	1Ch	878	279	9.96	
	1Dh	907	319	9.08	
	1Eh	933	357	8.35	
	1Fh	957	392	7.74	
1Bh	0Ah	821	182	13.09	
	0Bh	854	226	11.55	
	0Ch	885	268	10.38	
	0Dh	913	308	9.45	
	0Eh	939	346	8.68	
	0Fh	962	381	8.04	
	10h	983	415	7.50	
	11h	1,000	446	7.01	
	12h	1,016	475	6.59	
	13h	1,028	503	<b>6.22</b>	
	1Ah	826	188	12.86	
	1Bh	858	232	11.38	
	1Ch	885	268	10.38	
	1Dh	913	308	9.45	
	1Eh	939	345	8.69	
	1Fh	962	381	8.04	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
1Ch	0Ah	829	171	13.71	
	0Bh	861	215	12.06	
	0Ch	892	257	10.81	
	0Dh	919	297	9.82	
	0Eh	945	334	9.02	
	0Fh	967	370	8.35	
	10h	987	404	7.77	
	11h	1,004	435	7.27	
	12h	1,019	464	6.83	
	13h	1,031	491	<b>6.44</b>	
	19h	799	131	15.71	
	1Ah	833	177	13.46	
	1Bh	865	221	11.87	
	1Ch	891	257	10.81	
	1Dh	919	297	9.83	
	1Eh	944	334	9.02	
	1Fh	967	370	8.35	
1Dh	09h	802	114	16.95	
	0Ah	836	160	14.36	
	0Bh	868	204	12.58	
	0Ch	898	246	11.25	
	0Dh	925	286	10.20	
	0Eh	950	324	9.36	
	0Fh	972	359	8.65	
	10h	992	393	8.05	
	11h	1,008	424	7.52	
	12h	1,022	454	7.06	
	13h	1,034	481	6.65	
	19h	806	120	16.55	
	1Ah	841	166	14.09	
	1Bh	872	210	12.38	
	1Ch	898	246	11.25	
	1Dh	925	286	10.21	
	1Eh	950	323	9.36	
	1Fh	972	359	8.65	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

**Table 21-11. Tx Programmable Drive and De-Emphasis Levels (Cont.)**

<b>POST_CURSOR [4:0]</b>	<b>DRV_LVL [4:0]</b>	<b>Transition Amplitude (mV)</b>	<b>Non-Transition Amplitude (mV)</b>	<b>De-Emphasis (dB)</b>	<b>Notes</b>
1Eh	09h	809	103	17.90	
	0Ah	843	149	15.04	
	0Bh	875	193	13.11	
	0Ch	904	235	11.69	
	0Dh	931	275	10.59	
	0Eh	955	313	9.69	
	0Fh	977	349	8.95	
	10h	996	382	8.32	
	11h	1,012	413	7.77	
	12h	1,025	443	7.29	
	13h	1,037	470	6.87	
	19h	814	109	17.44	
	1Ah	848	155	14.74	
	1Bh	879	199	12.90	
	1Ch	904	235	11.70	
1Fh	1Dh	931	275	10.59	
	1Eh	955	313	9.70	
	1Fh	977	348	8.95	
	09h	817	93	18.91	
	0Ah	850	139	15.75	
	0Bh	881	183	13.66	
	0Ch	910	225	12.15	
	0Dh	936	265	10.98	
	0Eh	960	302	10.04	
	0Fh	981	338	9.26	
	10h	999	371	8.60	
	11h	1,015	403	8.03	
	12h	1,028	432	7.53	
	13h	1,039	459	7.09	
	19h	821	99	18.40	
	1Ah	854	145	15.43	
	1Bh	885	188	13.44	
	1Ch	910	225	12.15	
	1Dh	936	264	10.98	
	1Eh	960	302	10.04	
	1Fh	981	338	9.26	

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## 21.7.1 Default Transmit Settings

**Table 21-12** lists the default values of the Transmit Drive and Post-Cursor De-Emphasis levels (**SerDes Drive Level *x*** and **Post-Cursor Emphasis Level *x*** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **B98h**, **B9Ch**, **BA0h**, and **BA4h**, and offsets **BA8h**, **BACh**, **BB0h**, and **BB4h**, respectively)).

**Table 21-12. Default Transmit Settings**

Mode (dB)	Link Speed (GT/s)	DRV_LVL [4:0]	POST_CURSOR [4:0]	Transition Amplitude (mV)	Non-Transition Amplitude (mV)	Equalization <sup>a</sup> (dB)
-3.5	2.5	10h	0Dh	896	586	-3.69
-6	5.0	0Eh	15h	901	415	-6.73 <sup>b</sup>

a. *dB Equalization formula:*

$$20 \times \log[(\text{Non-Transition Amplitude}) / (\text{Transition Amplitude})]$$

b. *The -6 dB Setting is slightly larger than the maximum -6.5 dB specification, to better compensate for FR4 loss characteristics across a typical backplane application.*

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

## 21.8 Receive Characteristics

The Receiver circuit includes programmable equalization, to further compensate for the low-pass FR4 loss characteristics of the channel.

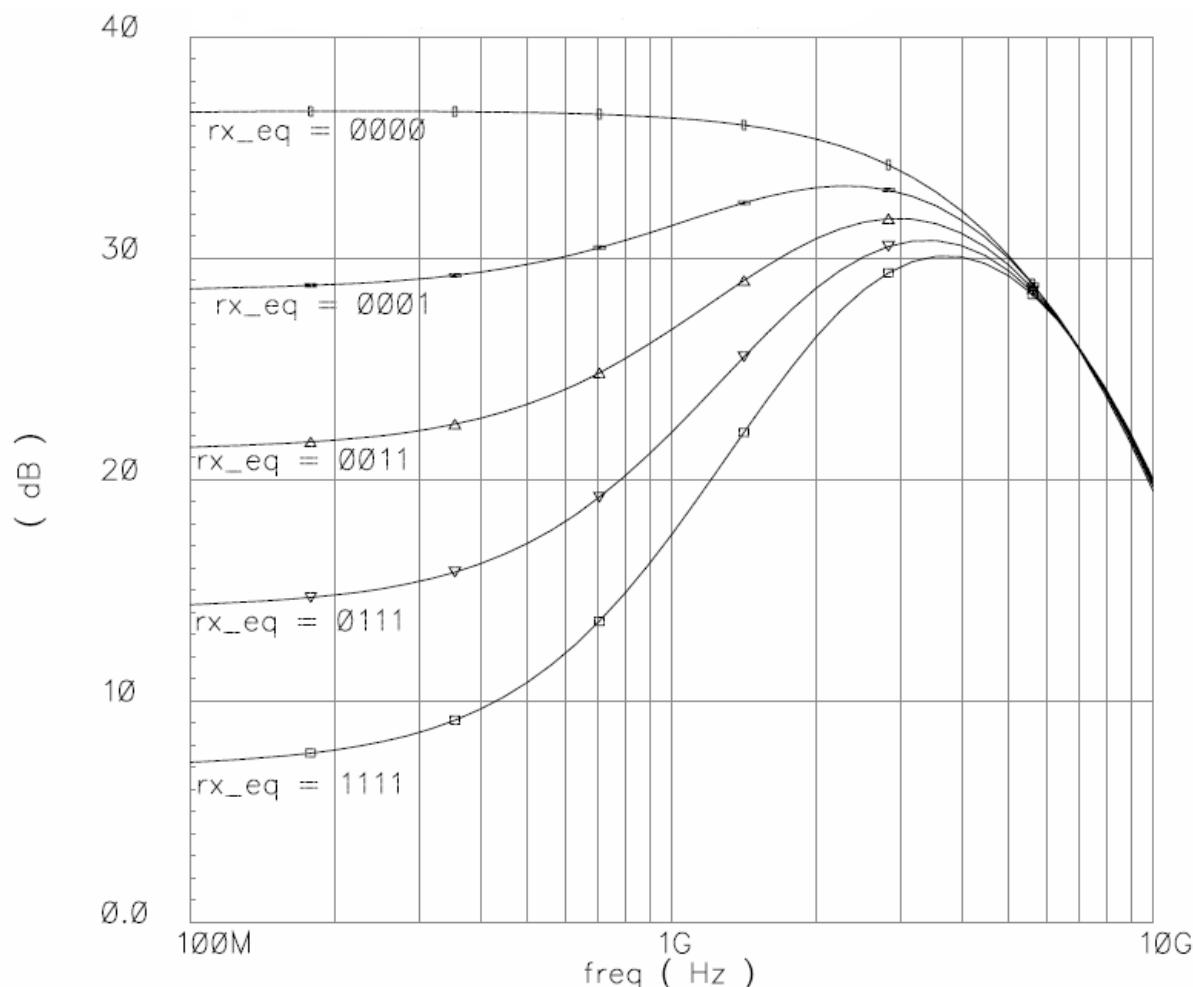
### 21.8.1 Receive Equalization

[Table 21-13](#) lists the programmable bits used for controlling the Receiver circuit's electrical characteristics, to mitigate the effects of signal loss and distortion across the PCB channel. The **Receiver Equalization Level x** registers (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offsets **BB8h** and **BBCh**) provide access to all 16 Lanes for Rx Equalization programmability. [Figure 21-2](#) illustrates the Rx Equalization frequency characteristics.

**Table 21-13. Receiver Equalization Settings**

Rx Equalization[3:0]	Equalization
0h (default)	Off
1h	Minimum
2h to 3h	Low
4h to 6h	Low to Medium
7h to 9h	Medium
Ah to Dh	High to Medium
Eh to Fh	Maximum

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

**Figure 21-2. Rx Equalization Frequency Characteristics**

## 21.8.2 Receiver Electrical Idle

The Receiver circuit contains a signal detect circuit that is used to detect signal idling at the input. The threshold to detect the idle level is programmable, using the **Signal Detect Level** register (Port 0, and also the NT Port Virtual Interface if Port 0 is a Legacy NT Port, offset **BC0h**). A value of 00b provides the lowest signal voltage level detection threshold.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3



## Chapter 22 Thermal and Mechanical Specifications

### 22.1 Thermal Characteristics

The PEX 8619 does not include a heat sink. The information described in this section is based upon sample thermal performance when a heat sink is used with the PEX 8619, and is provided for reference only.

#### 22.1.1 Sample Thermal Data

Table 22-1 lists sample thermal data for the PEX 8619 at Commercial temperature, as well as a recommendation for Industrial Temperature Range support.

**Table 22-1. Sample Thermal Data (19 x 19 mm<sup>2</sup> HSBGA Package)<sup>a, b</sup>**

Maximum Power (Watts) <sup>c</sup>	Heat Sink (Yes/No)	Airflow Velocity (m/s)	$\Theta_{JA}$ (°C/W)		$\Theta_{JC}$ (°C/W)	Heat Sink
			4-Layer JEDEC Board	8-Layer JEDEC Board		
3.70	No	0	16.21	11.53	4.78	N/A
		1	14.14	10.36		
		2	13.40	9.97		
	Yes	0	14.08	10.58		21 x 21 mm <sup>2</sup> and 11.4-mm-height heat sink, TIM chomerics T412, 0.23-mm thick and 1.41 W/mK Thermal conductivity <sup>d</sup>
		1	10.57	8.60		30 x 30 mm <sup>2</sup> and 15-mm height heat sink, TIM chomerics T412, 0.23-mm thick and 1.41 W/mK Thermal conductivity <sup>e</sup>
		2	9.16	7.77		
		2	N/A	6.80		

- a. The Maximum Operating Junction Temperature is 110°C.  
The Maximum Junction Temperature for Reliability is 125°C.
- b. The shaded fields provide a recommendation that allows the PEX 8619 to support Industrial Temperature Range (-40 to +85°C).
- c. The maximum power value listed assumes the conditions listed in Chapter 21, “Electrical Specifications,” at Gen 2 (5.0 GT/s).
- d. Heat Sink Part Number: Wakefield 624-60AB-T4.
- e. Heat Sink Part Number: Alpha Novatech W30-15W.

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 22.2 General Package Specifications

Table 22-2 lists general package specifications. For a more complete list, refer to [Figure 22-1](#).

**Table 22-2. General Package Specifications**

Parameter	Specification
Package Type	Heat Slug Ball Grid Array (HSBGA)
Quantity of Balls	324
Package Dimensions	19 x 19 mm <sup>2</sup> (approximately 1.71 ±0.19-mm high)
Ball Matrix Pattern	18 x 18 (6 x 6 mm <sup>2</sup> center area <b>reserved</b> for Ground)
Ball Pitch	1.00 mm
Ball Diameter	0.60 mm
Ball Spacing	0.40 mm

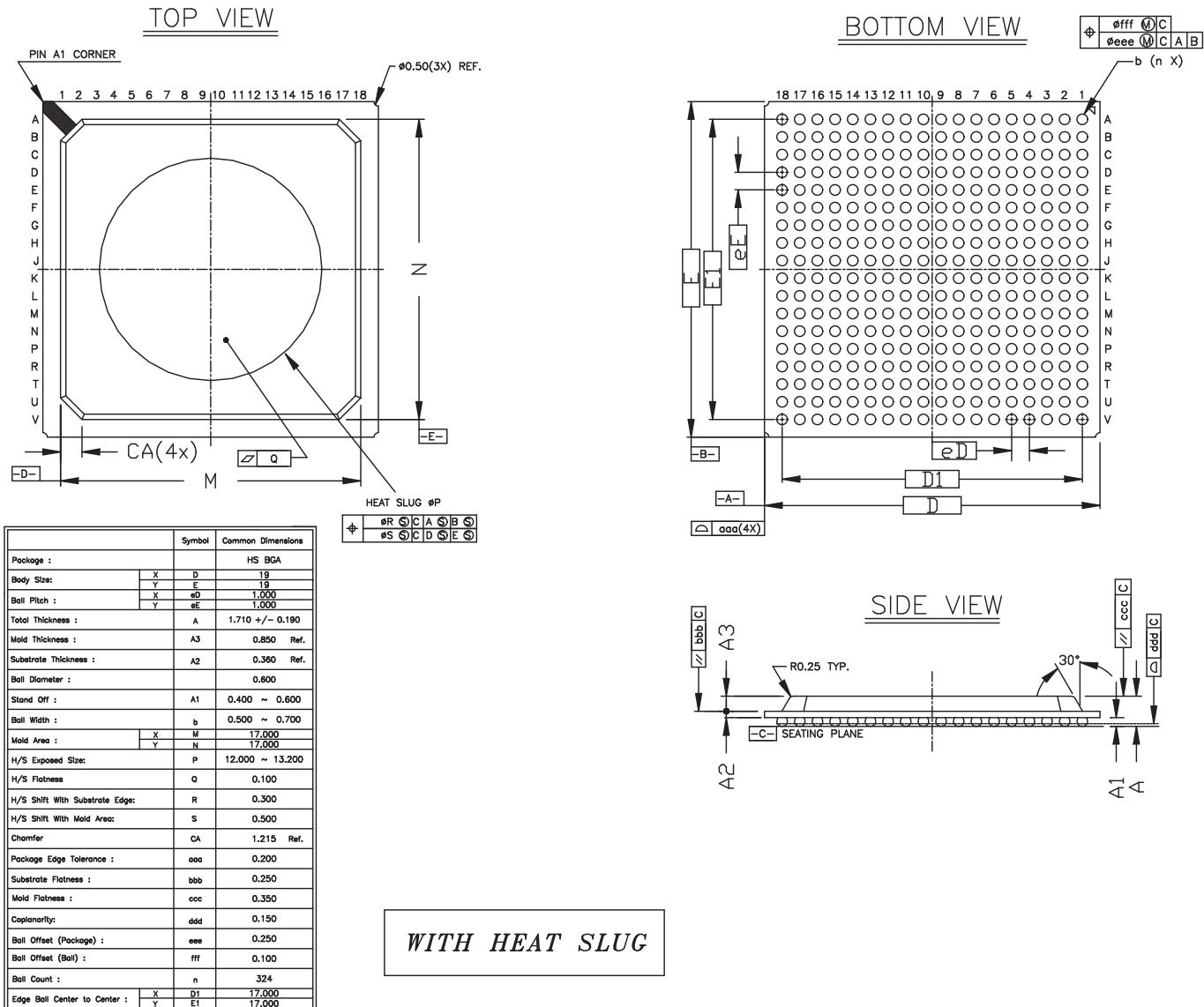
NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved

## 22.3 Mechanical Dimensions

**Figure 22-1. Mechanical Dimensions (19 x 19 mm<sup>2</sup> HSBGA Package)**



NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- kotlin-novator | vtuz vova**

ExpressLane PEX 8619-BA 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA Data Book, Version 1.3

Copyright © 2011 by PLX Technology, Inc. All Rights Reserved



## Appendix A General Information

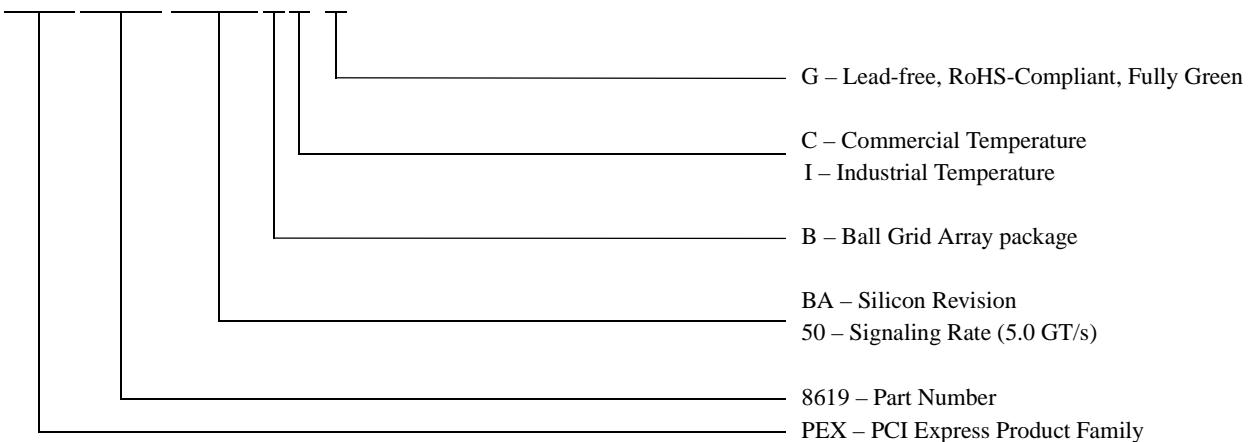
### A.1 Product Ordering Information

Contact your local [PLX Sales Representative](#) for ordering information.

**Table A-1. Product Ordering Information**

Part Numbers	Description
PEX8619-BA50BC	PEX 8619 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA HSBGA Leaded Package (19 x 19 mm <sup>2</sup> , 324-ball)
PEX8619-BA50BC G	PEX 8619 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA HSBGA Lead-Free Package (19 x 19 mm <sup>2</sup> , 324-ball)
PEX8619-BA50BI G	PEX 8619 16-Lane, 16-Port PCI Express Gen 2 Switch with DMA HSBGA Lead-Free Package (19 x 19 mm <sup>2</sup> , 324-ball) with Industrial Temperature Support

#### PEX8619-BA50BC G



PEX 8619BA-AIC4U4D Kit	PEX 8619 Rapid Development Kit with x4 Upstream Port and Four x4 Downstream Ports
PEX 8619BA-BB4U1D Kit	PEX 8619 Base Board Rapid Development Kit with x4 Upstream Port, Twelve x1 Downstream Ports, One x4 PCI Express Cable, Host Adapter Card
PEX 8619BA-BB8U1D Kit	PEX 8619 Base Board Rapid Development Kit with x8 Upstream Port, Eight x1 Downstream Ports, Two x4 PCI Express Cables, Host Adapter Card
x1 Adapter	PCI Express x8 to x1 Adapter
x4 Adapter	PCI Express x8 to x4 Adapter

NDA CONFIDENTIAL -- kotlin-novator | vtuz vova

## A.2 United States and International Representatives and Distributors

PLX Technology, Inc., representatives and distributors are listed at [wwwplxtech.com](http://wwwplxtech.com).

## A.3 Technical Support

PLX Technology, Inc., technical support information is listed at [wwwplxtech.com/support](http://wwwplxtech.com/support), or call 800 759-3735 (domestic only) or 408 774-9060.