Intel® Core™ i7 Processor Family for LGA2011 Socket

Datasheet – Volume 2 of 2

Supporting Desktop Intel® Core™ i7-4960X Extreme Edition Processor Series for the LGA2011 Socket

Supporting Desktop Intel® Core™ i7-49xx and i7-48xx Processor Series for the LGA2011 Socket

September 2013

329367-001

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm

Any software source code reprinted in this document is furnished under a software license and may only be used or copied in accordance with the terms of that license.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families: Go to: Learn About Intel Processor Numbers: http://www.intel.com/content/www/us/en/processors/processor-numbers.html

Requires a system with a 64-bit enabled processor, chipset, BIOS and software. Performance will vary depending on the specific hardware and software you use. Consult your PC manufacturer for more information. For more information, visit http://www.intel.com/info/em64t

WARNING: Altering clock frequency and/or voltage may: (i) reduce system stability and useful life of the system and processor; (ii) cause the processor and other system components to fail; (iii) cause reductions in system performance; (iv) cause additional heat or other damage; and (v) affect system data integrity. Intel has not tested, and does not warranty, the operation of the processor beyond its specifications. Intel assumes no responsibility that the processor, including if used with altered clock frequencies and/or voltages, will be fit for any particular purpose. For more information, visit Overclocking Intel Processors: http://www.intel.com/content/www/us/en/gaming/overclocking/overclocking-intel-processors.html

Intel® Virtualization Technology requires a computer system with an enabled Intel® processor, BIOS, and virtual machine monitor (VMM). Functionality, performance or other benefits will vary depending on hardware and software configurations. Software applications may not be compatible with all operating systems.Consult your PC manufacturer. For more information, visit http://www.intel.com/go/virtualization

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Hyper-Threading Technology requires a computer system with a processor supporting HT Technology and an HT Technology-enabled chipset, BIOS and operating system. Performance will vary depending on the specific hardware and software you use. For more information including details on which processors support HT Technology, see htp://www.intel.com/info/hyperthreading.

"Intel® Turbo Boost Technology requires a PC with a processor with Intel Turbo Boost Technology capability. Intel Turbo Boost Technology performance varies depending on hardware, software and overall system configuration. Check with your PC manufacturer on whether your system delivers Intel Turbo Boost Technology.For more information, see http://www.intel.com/technology/turboboost."

Enhanced Intel SpeedStep® Technology See the Processor Spec Finder or contact your Intel representative for more information.

No computer system can provide absolute security under all conditions. Intel® Trusted Execution Technology (Intel® TXT) requires a computer system with Intel® Virtualization Technology, an Intel TXT-enabled processor, chipset, BIOS, Authenticated Code Modules and an Intel TXT-compatible measured launched environment (MLE). The MLE could consist of a virtual machine monitor, an OS or an application. In addition, Intel TXT requires the system to contain a TPM v1.2, as defined by the Trusted Computing Group and specific software for some uses. For more information, see http://www.intel.com/technology/security

See the Processor Specification Finder at http://ark.intel.com/ or contact your Intel representative for more information.

Available on select Intel® Core™ processors. Requires an Intel® HT Technology-enabled system. Consult your PC manufacturer. Performance will vary depending on the specific hardware and software used. For more information including details on which processors support HT Technology, visit http://www.intel.com/info/hyperthreading

Intel, Enhanced Intel® SpeedStep® Technology, Intel® 64 Technology, Intel® Virtualization Technology (Intel® VT), Intel® VT-d, Intel® Turbo Boost Technology, Intel® Hyper-Threading Technology (Intel® HT Technology), Intel® Streaming SIMD Extensions (Intel® SSE), Intel Core, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others.

Contents

Figures

Tables

Revision History

Revision Number	Description	Date
001	• Initial release	September 2013

§

§

(This page intentionally left blank)

1 Introduction

This document is Volume 2 of 2 of the datasheet for the Intel® Core™ i7 processor family for LGA2011 Socket. Volume 2 provides register information for these processors.

Volume 2 of 2 describes the Configuration Status Registers (CSRs) of each individual functional block in Uncore logic. The processor contains one or more PCI devices within each individual functional block. CSRs are the basic hardware elements that configure the Uncore logic to support various system topologies, memory configuration, and densities.

The processor family contains one or more PCI devices within a single physical component. The configuration registers for these devices are mapped as devices residing on the PCI Bus assigned to the processor socket.

Volume 1 of 2 provides processor feature details, supported technologies, interface functional descriptions, power management descriptions, DC electrical specifications, land and signal definitions, and additional feature information pertinent to the implementation and operation of the processor. See the *Intel® Core™ i7 Processor Family for LGA2011 Socket* Datasheet, Volume 1 of 2 (see Related Documents section).

Note: Some processor features are not available on all platforms. Refer to the processor specification update document for details.

Note: Throughout this document, Intel® Core™ i7 processor family for LGA2011 Socket may be referred to as “processor”.

1.1 Document Terminology

Table 1-1. Processor Terminology (Sheet 1 of 2)

Terminology	Description
DDR3	Third generation Double Data Rate SDRAM memory technology that is the successor to DDR2 SDRAM
DMA	Direct Memory Access
DMI2	Direct Media Interface 2
DTS	Digital Thermal Sensor
Enhanced Intel® SpeedStep® Technology	Intel technology that allows the operating system to reduce power consumption when performance is not needed.
Execute Disable Bit	The Execute Disable bit allows memory to be marked as executable or non-executable–when combined with a supporting operating system. If code attempts to run in non-executable memory the processor raises an error to the operating system. This feature can prevent some classes of viruses or worms that exploit buffer overrun vulnerabilities and can help improve the overall security of the system. See the *Intel® 64 and IA-32 Architectures Software Developer's Manuals* for more detailed information.
Functional Operation	Refers to the normal operating conditions in which all processor specifications, including DC, AC, system bus, signal quality, mechanical, and thermal, are satisfied.
Home Agent (HA)	Responsible for memory transaction through the Ring and handles incoming/ outgoing memory transactions
Integrated Heat Spreader (IHS)	A component of the processor package used to enhance the thermal performance of the package. Component thermal solutions interface with the processor at the IHS surface.
Integrated Memory Controller (IMC)	The Memory Controller that is integrated on the processor die.
Intel® 64 Technology	64-bit memory extensions to the IA-32 architecture.
Intel® Turbo Boost Technology	Intel® Turbo Boost Technology is a way to automatically run the processor core faster than the marked frequency if the part is operating under power, temperature, and current specification limits of the Thermal Design Power (TDP). This results in increased performance of both single and multi-threaded applications.
Intel® Virtualization Technology (Intel® VT)	Processor virtualization which when used in conjunction with Virtual Machine Monitor software enables multiple, robust independent software environments inside a single platform.
Intel® VT-d	Intel® Virtualization Technology (Intel® VT) for Directed I/O. Intel® VT-d is a hardware assist, for enabling I/O device virtualization control (under system software (Virtual Machine Manager or OS). Intel® VT-d also enables robust security by providing protection from errant DMAs (by using DMA remapping, a key feature of Intel® VT-d).
IOV	I/O Virtualization
Jitter	Any timing variation of a transition edge or edges from the defined Unit Interval (UI).
LGA-2011 Socket	The 2011-land FC-LGA package mates with the system board through this surface mount, 2011-contact socket.
NCTF	Non-Critical To Function—NCTF locations are typically redundant ground or non-critical reserved, so the loss of the solder joint continuity at end of life conditions will not affect the overall product functionality.
PCH	Platform Controller Hub—The next generation chipset with centralized platform capabilities including the main I/O interfaces along with display connectivity, audio features, power management, manageability, security and storage features.
PCI Express* 3.0	The third generation PCI Express* specification that operates at twice the speed of PCI Express 2.0 (8Gb/s); however, PCI Express* 3.0 is completely backward compatible with PCI Express* 1.0 and 2.0.

Table 1-1. Processor Terminology (Sheet 2 of 2)

Terminology	Description
PCU	Power Control Unit
PECI	Platform Environment Control Interface
Processor	64-bit, single-core or multi-core component (package)
Processor Core	The term "processor core" refers to silicon die itself which can contain multiple execution cores. Each execution core has an instruction cache, data cache, and 256KB L2 cache. All execution cores share the L3 cache.
Rank	A unit of DRAM connecting four to eight devices in parallel. These devices are usually, but not always, mounted on a single side of a DDR3 DIMM.
Ring	Processor interconnect between the different Uncore modules
RP	Indicate Root Port for PCI Express
SCI	System Control Interrupt—Used in ACPI protocol.
SMBus	System Management Bus—A two-wire interface through which simple system and power management related devices can communicate with the rest of the system. It is based on the principals of the operation of the I^2C* two-wire serial bus from Philips* Semiconductor.
Intel® SSE	Intel® Streaming SIMD Extensions (Intel® SSE)
Storage Conditions	A non-operational state that the processor may be installed in a platform, in a tray, or loose. Processors may be sealed in packaging or exposed to free air. Under these conditions, processor landings should not be connected to any supply voltages, have any I/Os biased or receive any clocks. Upon exposure to "free air" (that is, unsealed packaging or a device removed from packaging material) the processor must be handled in accordance with moisture sensitivity labeling (MSL) as indicated on the packaging material.
TAC	Thermal Averaging Constant
TDP	Thermal Design Power
Uncore	The portion of the processor composed of the shared cache, IMC, and IIO.
Unit Interval	Signaling convention that is binary and unidirectional. In this binary signaling, one bit is sent for every edge of the forwarded clock, whether it be a rising edge or a falling edge. If a number of edges are collected at instances t_1, t_2, t_n,...., t_k then the UI at instance "n" is defined as: $UI_n = t_n - t_n - 1$
UR	Unsupported Requests
V_{CC}	Processor core power supply voltage
V_{SS}	Processor ground voltage
x1	Refers to a Link or Port with one physical lane
x16	Refers to a Link or Port with sixteen physical lanes
x4	Refers to a Link or Port with four physical lanes
x8	Refers to a Link or Port with eight physical lanes

1.2 Related Documents

Refer to the following documents for additional information.

Table 1-2. Processor Documents

Document	Document Number/ Location
Intel® Core™ i7 Processor Family for the LGA2011 Socket Datasheet, Volume 1	329366
Desktop Intel® Core™ i7 Processor Family for the LGA2011 Thermal Mechanical Specification and Design Guide	326199
Intel® Core™ i7 Processor Family for the LGA2011 Socket Specification Update	329368
Intel® X79 Express Chipset Datasheet	326200
Intel® X79 Express Chipset Thermal Mechanical Specifications and Design Guide	326202
Advanced Configuration and Power Interface Specification 3.0	http://www.acpi.info
PCI Local Bus Specification 3.0	http://www.pcisig.com/ specifications
PCI Express Base Specification - Revision 3.0	http://www.pcisig.com
DDR3 SDRAM Specification	http://www.jedec.org
Intel® 64 and IA-32 Architectures Software Developer's Manuals • Volume 1: Basic Architecture • Volume 2A: Instruction Set Reference, A-M • Volume 2B: Instruction Set Reference, N-Z • Volume 3A: System Programming Guide • Volume 3B: System Programming Guide *Intel® 64 and IA-32 Architectures Optimization Reference Manual*	http://www.intel.com/ products/processor/ manuals/index.htm
Intel® Virtualization Technology Specification for Directed I/O Architecture Specification	http://download.intel.com/ technology/computing/ vptech/ Intel(r)_VT_for_Direct_IO.pdf

§ §

2 Registers Overview and Configuration Process

This chapter covers

- Platform Configuration Structure
- Configuration Register Rules
- Register Terminology
- Notational Conventions

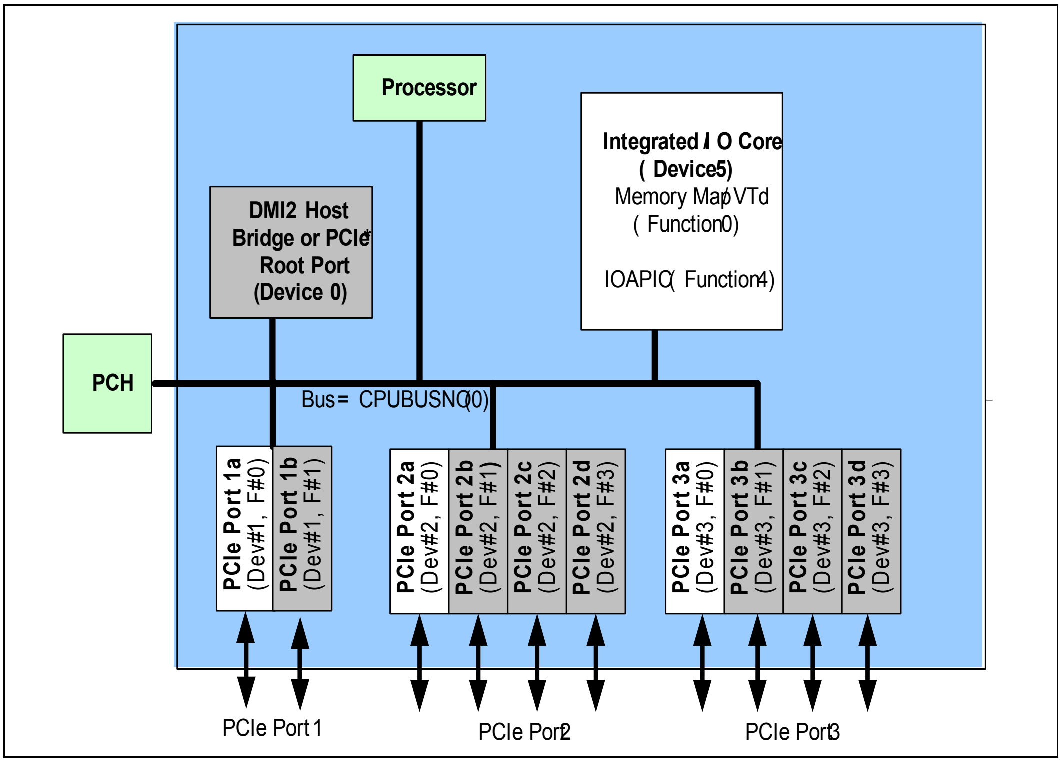
2.1 Platform Configuration Structure

The DMI2 physically connects the processor and the PCH. From a configuration standpoint the DMI2 is a logical extension of PCI Bus 0. DMI2 and the internal devices in the processor IIO and PCH logically constitute PCI Bus 0 to configuration software. As a result, all devices internal to the processor and the PCH appear to be on PCI Bus 0.

2.1.1 Processor IIO Devices (CPUBUSNO (0))

The processor IIO contains six PCI devices within a single, physical component. The configuration registers for the devices are mapped as devices residing on PCI Bus "CPUBUSNO(0)" where CPUBUSNO(0) is programmable by the BIOS.

Figure 2-1. Processor Integrated I/O Device Map

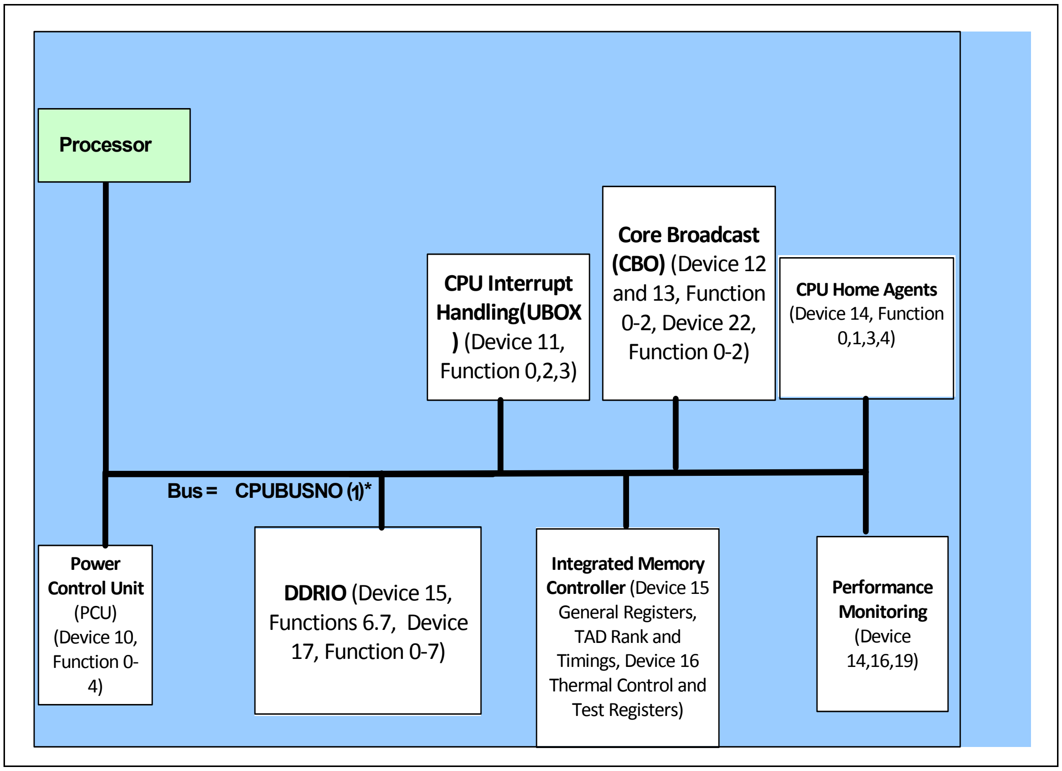

Processor
Integrated I/ O Core
(Device5)
Memory Map/VTd
(Function0)
IOAPIC(Function4)
DMI2 Host
Bridge or PCIe*
Root Port
(Device 0)
PCH
Bus= CPUBUSNO(0)
PCIe Port 1a (Dev#1, F#0)
PCIe Port 1b (Dev#1, F#1)
PCIe Port 2a (Dev#2, F#0)
PCIe Port 2b (Dev#2, F#1)
PCIe Port 2c (Dev#2, F#2)
PCIe Port 2d (Dev#2, F#3)
PCIe Port 3a (Dev#3, F#0)
PCIe Port 3b (Dev#3, F#1)
PCIe Port 3c (Dev#3, F#2)
PCIe Port 3d (Dev#3, F#3)
PCIe Port 1
PCIe Port2
PCIe Port3

- **Device 0:** DMI2 Root Port. Logically this appears as a PCI device residing on PCI Bus 0. Device 0 contains the standard PCI header registers, extended PCI configuration registers and DMI2 device specific configuration registers.
- **Device 1:** PCI Express* Root Port 1a, 1b. Logically this appears as a "virtual" PCI-to-PCI bridge residing on PCI Bus 0 and is compliant with *PCI Express* Local Bus Specification Revision 2.0*. Device 1 contains the standard PCI Express*/PCI configuration registers including PCI Express* Memory Address Mapping registers. It also contains the extended PCI Express configuration space that include PCI Express error status/control registers and Virtual Channel controls.
- **Device 2**: PCI Express* Root Port 2a, 2b, 2c, and 2d. Logically this appears as a "virtual" PCI-to-PCI bridge residing on PCI bus 0 and is compliant with *PCI Express* Specification Revision 2.0*. Device 2 contains the standard PCI Express*/PCI configuration registers including PCI Express* Memory Address Mapping registers. It also contains the extended PCI Express* configuration space that include PCI Express* Link status/control registers and Virtual Channel controls.
- **Device 3:** PCI Express Root Port 3a, 3b, 3c, and 3d. Logically this appears as a "virtual" PCI-to-PCI bridge residing on PCI Bus 0 and is compliant with *PCI Express* Local Bus Specification Revision 2.0*. Device 3 contains the standard PCI Express*/PCI configuration registers including PCI Express* Memory Address Mapping registers. It also contains the extended PCI Express configuration space that include PCI Express* error status/control registers and Virtual Channel controls.
- **Device 5:** Integrated I/O Core. This device contains the Standard PCI registers for each of its functions. This device implements three functions; Function 0 contains Address Mapping, Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d) related registers and other system management registers. Function 1 contains PCIe*. Function 4 contains System Control/Status registers and miscellaneous control/status registers on power management and throttling.

2.1.2 Processor Uncore Devices (CPUBUSNO (1))

The configuration registers for these devices are mapped as devices residing on the PCI bus assigned to the processor socket. Bus number is derived by the maximum bus range setting and processor socket number.

Figure 2-2. Processor Uncore Devices Map


Processor
CPU Interrupt Handling(UBOX) (Device 11, Function 0,2,3)
Core Broadcast (CBO) (Device 12 and 13, Function 0-2, Device 22, Function 0-2)
CPU Home Agents (Device 14, Function 0,1,3,4)
Bus = CPUBUSNO (1)*
Power Control Unit (PCU) (Device 10, Function 0-4)
DDRIO (Device 15, Functions 6.7, Device 17, Function 0-7)
Integrated Memory Controller (Device 15 General Registers, TAD Rank and Timings, Device 16 Thermal Control and Test Registers)
Performance Monitoring (Device 14,16,19)

- **Device 10:** Processor Power Control Unit. Device 10, Function 0-4 contains the configurable PCU registers.
- **Device 11:** Processor Interrupt Event Handling (UBox). Device 11, Function 0 contains the processor Interrupt Control Registers. Device 11, Function 3 contains the Semaphore and Scratchpad Configuration Registers.
- **Device 12:** Processor Core Broadcast. Device 12, Function 0-7 contains the Unicast Configuration Registers.
- **Device 13:** Processor Core Broadcast. Device 13, Function 0-6 contains the Unicast configuration registers
- **Device 14:** Processor Home Agent 0. Device 14, Function 0 contains the processor Home Agent Target Address Configuration Registers for the Memory Controller. Device 14, Function 1 contains processor Home Agent Performance Monitoring Registers.
- **Device 15:** Integrated Memory Controller 0. Device 15, Function 0 contains the General and MemHot Registers for Integrated Memory Controller 0 and resides. Device 15, Function 2-5 contains the Target Address Decode, Channels Rank, and Memory Timing Registers. Device 15, Functions 6-7 contains DDRIO registers.
- **Device 16:** Integrated Memory Controller 1. Device 16, Function 0, 1, 4 and 5 contains the Thermal control registers for Integrated Memory Controller 1 Channel 0, Channel 1, Channel 2, Channel 3. Device 16, Function 2, 3 and 7 contains the test registers for the Integrated Memory Controller 1.
- **Device 17:** DDRIO. Device 17, Function 0 to 3 address DDR Channels 2 and 3, while the registers in Function 4 to 7 addresses DDR Channels 0 and 1.

- **Device 19:** Processor Performance Monitoring and Ring. Device 19 Function 0 contains the processor ring to PCI Express agent. Device 19, Function 1 contains the processor Ring to PCI Express performance monitoring registers. Device 19, Function 4 -6 contains the processor performance monitoring registers.
- **Device 22:** Processor Core Broadcast. Device 22 Function 1-2 contains the Caching agent broadcast configuration registers for the Memory Controller. Device 22 Function 0 contains the System Address Decode Registers.

2.2 Configuration Register Rules

The processor supports the following configuration register types:

- PCI Configuration Registers (CSRs): CSRs are chipset specific registers that are located at PCI defined address space.
- Machine Specific Registers (MSRs): MSRs are machine specific registers that can be accessed by specific read and write instructions. The registers are OS ring 0 and BIOS accessible.
- Memory-Mapped I/O Registers: These registers are mapped into the system memory map as MMIO low or MMIO high. The registers are accessed by any operating system driver running on the platform. This register space is introduced with the integration of some of the chipset functionality.

2.2.1 Configuration Space Registers (CSR) Access

Configuration Space Registers are accessed by means of the configuration transaction mechanism defined in the PCI specification. It uses the bus:device:function number concept to address a specific device's configuration space.

All configuration space register access occurs over the Message Channel through UBox but can also come from a variety of different sources such as:

- Local cores
- PECI or JTAG

Configuration space registers can be read or written in Byte, WORD (16-bit), or Dword (32-bit) quantities. *Accesses larger than a Dword to PCI Express configuration space will result in unexpected behavior.* All multi-byte numeric fields use "little-endian" ordering (that is, lower addresses contain the least significant parts of the field).

2.2.1.1 PCI Bus Number

In the tables shown for IIO devices (0 – 7), the PCI Bus numbers are all marked as "Bus 0". The specific bus number for all PCIe* devices in the processor is specified in the CPUBUSNO register which exists in the I/O module's configuration space. Bus number is derived by the maximum bus range setting and processor socket number.

2.2.1.2 Uncore Bus Number

In the tables shown for Uncore devices (8 – 19), the PCI Bus numbers are all marked as "Bus 1". This means that the actual bus number is CPUBUSNO(1); where CPUBUSNO(1) is programmable by BIOS depending on which socket is used. The specific bus number for all PCIe* devices in the processor is specified in the CPUBUSNO register.

2.2.1.3 Device Mapping

Each component in the processor is uniquely identified by a PCI bus address consisting of Bus Number, Device Number, and Function Number. Device configuration is based on the PCI Type 0 configuration conventions. All processor registers appear on the PCI bus assigned for the processor socket. Bus number is derived by the maximum bus range setting and processor socket number.

Table 2-1. Functions Specifically Handled by the Processor (Sheet 1 of 2)

Register Group	DID	Device	Function	Comment
DMI2	E00h	0	0	x4 Link from Processor to PCH
PCI Express Root Port in DMI2 Mode	E01h	0	0	Device 0 will work as a x4 PCI Express Port
PCI Express Root Port 2	E04h, E05h, E06h, E07h	2	0 -3	x16, x8 or x4 maximum link width
PCI Express Root Port 3	E08, E09h, EOAh, E0Bh	3	0-3	x16, x8, or x4 maximum link width
Core	E28h	5	0	Address Map, VTd_Misc, System Management
	E29h	5	1	Hot-Plug
	E2Ah	5	2	RAS, Control Status and Global Errors
	E2Ch	5	4	I/O APIC
PCU	EC0h, EC1h, EC2h EC3h EC4h	10	0-4	Power Control Unit
UBOX	E1Eh	11	0	Scratchpad and Semaphores
	E7Dh	11	2	Scratchpad and Semaphores
	E1F	11	3	Scratchpad and Semaphores
Caching Agent (CBo)	EE0h, EE2h, EE4h, EE6h, EE8h, EEAh, EECh, EEEh	12	0-7	Unicast Registers
	EE1h, EE3h, EE5h, EE7h, EE9h, EEBh, EEDh	13	0-6	Unicast Registers
	EC8h	22	0	System Address Decoder
	EC9h, ECAh	22	1-2	Broadcast Registers
Home Agent (HA) 0	EA0h, E30h	14	0-1	Processor Home Agent 0
Integrated Memory Controller 0	EA8h	15	0	General and Memhot registers
	EAAh, EABh, EACh, EADh	15	2 -5	Target Address Decoder Registers

Table 2-1. Functions Specifically Handled by the Processor (Sheet 2 of 2)

Register Group	DID	Device	Function	Comment
Integrated Memory Controller 1	EB2 EB3 EB7	16	2,3,7	Test Registers
	EB0, EB1, EB4 EB5	16	0,1,4,5	Channel 0-3 Thermal Control Registers
DDRIO	EF8h, EF9h, EFAh, EFBh, EFCh, EFDh	17	0 -7	DDRIO Channel 0-3
R2PCIe	E1Dh	19	0	R2PCIE
	E34h	19	1	PCI Express Ring Performance Monitoring

2.2.1.4 Unimplemented Devices / Functions and Registers

Configuration reads to unimplemented functions and devices will return all ones emulating a master abort response.

Note: There is no asynchronous error reporting that happens when a configuration read master aborts. Configuration writes to unimplemented functions and devices will return a normal response.

Software should not attempt or rely on reads or writes to unimplemented registers or register bits. Unimplemented registers should return all zeroes when read. Writes to unimplemented registers are ignored. For configuration writes to these register (require a completion), the completion is returned with a normal completion status (not master-aborted).

2.2.1.5 Device Hiding

The processor provides a mechanism by which various PCI devices or functions within the unit can be hidden from the host configuration software. This mechanism is needed in cases where a device or function is not used or is available for use.

Note: This is because either the device is turned off by a fuse or the device is not serving any meaningful purpose in a given platform configuration.

This hiding mechanism is implemented by means of the DEVHIDE register:

- Devices that are hidden from host configuration space by means of the DEVHIDE register are not hidden from the configuration space as seen from the JTAG/SMBus port of an IIO. All PCI devices are always visible by means of the JTAG/SMBus.
- Devices or functions when turned off by respective fuses are always hidden (and not programmable to be unhidden) from host configuration space and also from PECI/JTAG.
- Devices that are not turned off by respective fuses, but are otherwise not used in a given platform configuration can be hidden from host configuration space by BIOS appropriately programming the DEVHIDE register.
- The only change DEVHIDE register makes is to abort Type0 configuration accesses to the device space itself.

2.2.2 Memory-Mapped I/O Registers

The PCI standard provides not only configuration space registers but also registers which reside in memory-mapped space. For PCI devices, this is typically where the majority of the driver programming occurs and the specific register definitions and characteristics are provided by the device manufacturer. Access to these registers are typically accomplished by means of processor reads and writes to non-coherent (UC) or write-combining (WC) space.

The processor has relatively few of these; however, the integration of some of the chipset functionality has brought with it some I/O devices. These devices include Memory-Mapped I/O registers.

Reads and writes to memory-mapped registers can be accomplished with 1, 2, 4, or 8 byte transactions.

2.3 Register Terminology

The bits in configuration register descriptions will have an assigned attribute from the following table. Bits without a Sticky attribute are set to their default value by a hard reset.

Note: Table 2-2 is a comprehensive list of all possible attributes and included for completeness.

Table 2-2. Register Attributes Definitions (Sheet 1 of 2)

Attribute	Description
RO	**Read Only:** These bits can only be read by software, writes have no effect. The value of the bits is determined by the hardware only.
RW	**Read/Write:** These bits can be read and written by software.
RC	**Read Clear Variant:** These bits can be read by software, and the act of reading them automatically clears them. Hardware is responsible for writing these bits, and therefore the -V modifier is implied.
W1S	**Write 1 to Set:** Writing a 1 to these bits will set them to 1. Writing 0 will have no effect. Reading will return indeterminate values and read ports are not requited on the register. These are not supported by critter, and today is only allowed in the Cbo.
WO	**Write Only:** These bits can only be written by microcode, reads return indeterminate values. Microcode that wants to ensure this bit was written must read wherever the side-effect takes place.
RW-O	**Read/Write Once:** These bits can be read by software. After reset, these bits can only be written by software once, after which the bits becomes 'Read Only'.
RW-L	**Read/Write Lock:** These bits can be read and written by software. Hardware can make these bits 'Read Only' by means of a separate configuration bit or other logic.
RW1C	**Read/Write 1 to Clear:** These bits can be read and cleared by software. Writing a '1' to a bit clears it, while writing a '0' to a bit has no effect.
ROS	**RO Sticky:** These bits can only be read by software, writes have no effect. The value of the bits is determined by the hardware only. These bits are only re-initialized to their default value by a PWRGOOD reset.
RW1S	**Read, Write 1 to Set:** These bits can be read. Writing a '1' to a given bit will set it to 1. Writing a '0' to a given bit will have no effect. It is not possible for software to set a bit to '0'. The 1->0 transition can only be performed by hardware. These registers are implicitly -V.
RWS	**R/W Sticky:** These bits can be read and written by software. These bits are only re-initialized to their default value by a PWRGOOD reset.

Table 2-2. Register Attributes Definitions (Sheet 2 of 2)

Attribute	Description
RW1CS	**R/W1C Sticky:** These bits can be read and cleared by software. Writing a '1' to a bit clears it, while writing a '0' to a bit has no effect. These bits are only re-initialized to their default value by a PWRGOOD reset.
RW-LB	**Read/Write Lock Bypass**: Similar to RW-L, these bits can be read and written by software. Hardware can make these bits "Read Only" by means of a separate configuration bit or other logic. However, RW-LB is a special case where the locking is controlled by the lock-bypass capability that is controlled by the lock-bypass enable bits. Each lock-bypass enable bit enables a set of config request sources that can bypass the lock. The requests sourced from the corresponding bypass enable bits will be lock-bypassed (that is, RW) while requests sourced from other sources are under.
RO-FW	**Read Only Forced Write:** These bits are read only from the perspective of the cores. However, Pcode is able to write to these registers.
RWS-O	If a register is both sticky and "once" then the sticky value applies to both the register value and the "once" characteristic. Only a PWRGOOD reset will reset both the value and the "once" so that the register can be written to again.
RW-V	These bits may be modified by hardware. Typically, this is occurs based on values from hardware configuration straps for functions such as DMI2 and PCIe* I/O configuration. They also could be changed based on status or modes within internal state machines. Software cannot expect the values to stay unchanged. This is similar to "volatile" in software land.
RWS-L	If a register is both sticky and locked, then the sticky behavior only applies to the value. The sticky behavior of the lock is determined by the register that controls the lock.
RV	**Reserved:** These bits are reserved for future expansion and their value must not be modified by software. When writing these bits, software must preserve the value read.

2.4 Notational Conventions

2.4.1 Hexadecimal and Binary Numbers

Base 16 numbers are represented by a string of hexadecimal digits followed by the character h (for example, F82Eh). A hexadecimal digit is a character from the following set: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. Hexadecimal numbers can also be shown using a "0x" character preceding the number (for example 0x2A).

Base 2 (binary) numbers are represented by a string of 1s and 0s, sometimes followed by the character B (for example, 101B). The "B" designation is only used in situations where confusion as to the type of the number might arise.

Base 10 numbers are represented by a string of decimal digits followed by the character D (for example, 23D). The "D" designation is only used in situations where confusion as to the type of the number might arise.

§ §

3 Integrated Memory Controller (IMC) Configuration Registers

3.1 Device 15 Function 0

Table 3-1. IMC Device 15 Function 0 Register Address Map (Sheet 1 of 2)

Register Name	Offset	Size
pxpcap	0x40	32
mcmtr	0x7c	32
tadwayness_0	0x80	32
tadwayness_1	0x84	32
tadwayness_2	0x88	32
tadwayness_3	0x8c	32
tadwayness_4	0x90	32
tadwayness_5	0x94	32
tadwayness_6	0x98	32
tadwayness_7	0x9c	32
tadwayness_8	0xa0	32
tadwayness_9	0xa4	32
tadwayness_10	0xa8	32
tadwayness_11	0xac	32
mcmtr2	0xb0	32
mc_init_state_g	0xb4	32
rcomp_timer	0xc0	32
mh_maincntl	0x104	32
mh_sense_500ns_cfg	0x10c	32
mh_dtycyc_min_asrt_cntr_0	0x110	32
mh_dtycyc_min_asrt_cntr_1	0x114	32
mh_io_500ns_cntr	0x118	32
mh_chn_astn	0x11c	32
mh_temp_stat	0x120	32
mh_ext_stat	0x124	32
smb_stat_0	0x180	32
smbcmd_0	0x184	32
smbcntl_0	0x188	32
smb_tsod_poll_rate_cntr_0	0x18c	32
smb_stat_1	0x190	32
smbcmd_1	0x194	32
smbcntl_1	0x198	32

Table 3-1. IMC Device 15 Function 0 Register Address Map (Sheet 2 of 2)

Register Name	Offset	Size
smb_tsod_poll_rate_cntr_1	0x19c	32
smb_period_cfg	0x1a0	32
smb_period_cntr	0x1a4	32
smb_tsod_poll_rate	0x1a8	32

3.1.1 pxpcap

PCI Express* Capability

Type: CFG Bus: 1 Offset: 0x40		PortID: N/A Device: 15	Function: 0
Bit	**Attr**	**Default**	**Description**
29:25	RO	0x0	Interrupt Message Number (interrupt_message_number): N/A for this device
24:24	RO	0x0	Slot Implemented (slot_implemented): N/A for integrated endpoints
23:20	RO	0x9	Device/Port Type (device_port_type): Device type is Root Complex Integrated Endpoint
19:16	RO	0x1	Capability Version (capability_version): PCI Express* Capability is Compliant with Version 1.0 of the PCI Express* Specification. **Note:** This capability structure is not compliant with Versions beyond 1.0, since they require additional capability registers to be reserved. The only purpose for this capability structure is to make enhanced configuration space available. Minimizing the size of this structure is accomplished by reporting version 1.0 compliancy and reporting that this is an integrated root port device. As such, only three Dwords of configuration space are required for this structure.
15:8	RO	0x0	Next Capability Pointer (next_ptr): Pointer to the next capability. Set to 0 to indicate there are no more capability structures.
7:0	RO	0x10	Capability ID (capability_id): Provides the PCI Express* capability ID assigned by PCI-SIG.

3.1.2 mcmtr

MC Memory Technology

Type: CFG Bus: 1 Offset: 0x7c		PortID: N/A Device: 15	Function: 0
Bit	**Attr**	**Default**	**Description**
14:14	RW-LB	0x0	RSVD:
13:12	RW-LB	0x0	IMC_MODE (imc_mode): Memory mode: 00: Native DDR3 01: Reserved 10: Reserved 11: Reserved
11:10	RW-LB	0x0	CPGC_IOSAV (trng_mode): 00: IOSAV mode 01: Reserved 10: Reserved 11: Normal Mode
9:9	RW-LB	0x0	Enabling the bank xor address mapping (bank_xor_enable): When set, this bit will enable bank XOR'ing. This is targeted at workloads that bank thrashing caused by certain stride or page mappings. If one detects unexpectedly poor page hit rates, one can attempt to flip this bit to see if it helps. 0: Our base configuration. Bank selection is done using rank address bits 12:17:18 for open page mapping and bits 6:7:8 for close page mapping. 1: Bank XOR'ing enabled. Bank selection is done using rank address bits: (12^19): (17^20): (18^21) for open page mapping (6^19): (7^20): (8^21) for close page mapping
8:8	RW-LB	0x0	NORMAL (normal): 0: Training mode 1: Normal Mode
3:3	RW_LBV	0x0	DIR_EN (dir_en): If the directory disable fuse is set to directory disable state, this register bit is set to Read-Only (RO) with 0 value, that is directory is disabled. **Note:** This bit will only work if the SKU is enabled for this feature.
2:2	RW_LBV	0x0	ECC_EN (ecc_en): ECC enable. **Note:** This bit will only work if the SKU is enabled for this feature. DISECC will force override this bit to 0.
1:1	RW_LBV	0x0	LS_EN (ls_en): Use lock-step channel mode if set; otherwise, independent channel mode. This field should only be set for native ddr3 lockstep. **Note:** This bit will only work if the SKU is enabled for this feature.
0:0	RW-LB	0x0	CLOSE_PG (close_pg): Use close page address mapping if set; otherwise, open page.

3.1.3 tadwayness_[0:11]

TAD Range Wayness, Limit and Target.

There are total of 12 TAD ranges (N + P + 1 = number of TAD ranges; P = how many times channel interleave changes within the SAD ranges.).

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x80, 0x84, 0x88, 0x8c, 0x90, 0x94, 0x98, 0x9c, 0xa0, 0xa4, 0xa8, 0xac**				

Bit	Attr	Default	Description
31:12	RW-LB	0x0	TAD_LIMIT (tad_limit): highest address of the range in system address space, 64MB granularity, that is, TADRANGLIMIT[45:26].
11:10	RW-LB	0x0	TAD_SKT_WAY (tad_skt_way): socket interleave wayness 00 = 1 way, 01 = 2 way, 10 = 4 way, 11 = 8 way.
9:8	RW-LB	0x0	TAD_CH_WAY (tad_ch_way): channel interleave wayness 00 - interleave across 1 channel 01 - interleave across 2 channels 10 - interleave across 3 channels 11 - interleave across 4 channels This parameter effectively tells iMC how much to divide the system address by when adjusting for the channel interleave. Since both channels in a pair store every line of data, divide by 1 when interleaving across one pair and 2 when interleaving across two pairs. For HA, it tells how may channels to distribute the read requests across. When interleaving across 1 pair, this distributes the reads to two channels, when interleaving across 2 pairs, this distributes the reads across 4 pairs. Writes always go to both channels in the pair when the read target is either channel.
7:6	RW-LB	0x0	TAD_CH_TGT3 (tad_ch_tgt3): target channel for channel interleave 3 (used for 4-way TAD interleaving). This register is used in the iMC only for converting a rank address back to a system address.
5:4	RW-LB	0x0	TAD_CH_TGT2 (tad_ch_tgt2): target channel for channel interleave 2 (used for 3/4-way TAD interleaving).
3:2	RW-LB	0x0	TAD_CH_TGT1 (tad_ch_tgt1): target channel for channel interleave 1 (used for 2/3/4-way TAD interleaving).
1:0	RW-LB	0x0	TAD_CH_TGT0 (tad_ch_tgt0): target channel for channel interleave 0 (used for 1/2/3/4-way TAD interleaving).

3.1.4 mcmtr2

MC Memory Technology Register 2

Type: CFG **Bus:** 1 **Offset:** 0xb0		**PortID:** N/A **Device:** 15	**Function:** 0
Bit	**Attr**	**Default**	**Description**
3:0	RW-L	0x0	MONROE_CHN_FORCE_SR (monroe_chn_force_sr): Monroe Technology software channel force SRcontrol. When set, the corresponding channel is ignoring the ForceSRExit. A new transaction arrive at this channel will still cause the SR exit.

3.1.5 mc_init_state_g

Initialization state for boot, training and IOSAV.

Type: CFG **Bus:** 1 **Offset:** 0xb4		**PortID:** N/A **Device:** 15	**Function:** 0
Bit	**Attr**	**Default**	**Description**
12:9	RWS_L	0x0	cs_oe_en:
8:8	RWS_L	0x1	MC is in SR (safe_sr): This bit indicates if it is safe to keep the MC in SR during MC-reset. If it is clear when reset occurs, it means that the reset is without warning and the DDR-reset should be asserted. If set when reset occurs, it indicates that DDR is already in SR and it can keep it this way. This bit can also indicate MRC if reset without warning has occurred, and if it has, cold-reset flow should be selected. BIOS need to clear this bit at MRC entry.
7:7	RW-L	0x0	MRC_DONE (mrc_done): This bit indicates the PCU that the MRC is done, MC is in normal mode, ready to serve MRC should set this bit when MRC is done, but it does not need to wait until training results are saved in BIOS flash.
6:6	RW-L	0x0	Micro Break Point Synchronization (ubp_sync):
5:5	RW-L	0x1	DDRIO Reset (internal logic) (reset_io): Training Reset for DDRIO. It goes to both the left and right DDRIO blocks on MC on the platform and only the left side DDRIO block on Brickland
4:4	RW-L	0x1	IOSAV sequence channel sync (sync_iosav): This bit is used in order to sync the IOSAV operation in four channels. It is expect the BIOS to clear the bit after IOSAV test. Clearing the bit during test may lead to unknown behavior. By setting it four channels get the enable together.
3:3	RW-L	0x0	Refresh Enable (refresh_enable): If cold reset, this bit should be set by BIOS after 1) Initializing the refresh timing parameters 2) Running DDR through reset ad INIT sequence. If warm reset or S3 exit, this bit should be set immediately after SR exit.

Type: CFG **PortID:** N/A
Bus: 1 **Device:** 15 **Function:** 0
Offset: 0xb4

Bit	Attr	Default	Description
2:2	RW-L	0x0	DCLK Enable (for all channels) (dclk_enable):
1:1	RW-L	0x1	DDR_RESET (ddr_reset): DDR reset for all DIMM's from all channels within this socket. No IMC/DDRIO logic is reset by asserting this register. **Note:** This bit is negative logic. (that is, writing 0 to induce a reset and write 1 for not reset.)
0:0	RWS_L	0x0	Power-up MRC has completed successfully (pu_mrc_done):

3.1.6 rcomp_timer

RCOMP wait timer.

Defines the time from I/O starting to run RCOMP evaluation until RCOMP results are definitely ready. This counter is added in order to keep determinism of the process if operated in different mode.

This register also indicates that first RCOMP has been done.

Type: CFG **PortID:** N/A
Bus: 1 **Device:** 15 **Function:** 0
Offset: 0xc0

Bit	Attr	Default	Description
31:31	RW-V	0x0	rcomp_in_progress:
21:21	RW	0x0	ignore_mdll_locked_bit:
20:20	RW	0x0	no_mdll_fsm_override:
16:16	RW_LV	0x0	First RCOMP has been done in DDRIO (first_rcomp_done): This is a status bit that indicates the first RCOMP has been completed. It is cleared on reset, and set by MC hardware when the first RCOMP is completed. BIOS should wait until this bit is set before executing any DDR command.
15:0	RW	0xc00	COUNT (count): DCLK cycle count that MC needs to wait from the point it has triggered RCOMP evaluation until it can trigger the load to registers.

3.1.7 mh_maincntl

MEMHOT Main Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x104**				

Bit	Attr	Default	Description
18:18	RW	0x0	MHOT_EXT_SMI_EN (mhot_ext_smi_en): Generate SMI event when either MEM_HOT[1:0]# is externally asserted.
17:17	RW	0x0	MHOT_SMI_EN (mhot_smi_en): Generate SMI during internal MEM_HOT# event assertion
16:16	RW	0x0	Enabling external MEM_HOT sensing logic (mh_sense_en): Externally asserted MEM_HOT sense control enable bit. When set, the MEM_HOT sense logic is enabled.
15:15	RW	0x1	Enabling mem_hot output generation logic (mh_output_en): MEMHOT output generation logic enable control. When 0, the MEM_HOT output generation logic is disabled, that is, MEM_HOT[1:0]# outputs are in de-asserted state, no assertion regardless of the memory temperature. Sensing of externally asserted MEM_HOT[1:0]# is not affected by this bit. iMC will always reset the MH1_DIMM_VAL and MH0_DIMM_VAL bits in the next DCLK so there is no impact to the PCODE update to the MH_TEMP_STAT registers. When 1, the MEM_HOT output generation logic is enabled.
14:12	RW	0x6	MEM_HOT DUTY CYCLE RATE CONTROL DIVIDER (mh_duty_cyc_rate_cntl): Controlling the MEMHOT decrement counter rate. This field defines the number of bits to be right-shifted from the MH_DUTY_CYCLE_PRD value. When MH_DUTY_CYC_RATE_CNTL 0, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/1 1, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/2 2, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/4 3, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/8 4, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/16 5, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/32 6, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/64 7, the MH_DUTY_CYC_RATE = MH_DUTY_CYC_PRD/128
11:8	RW	0x0	MH_DUTY_CYC_RATE (mh_duty_cyc_rate): MEM_HOT decrement counter rate control field. The delta temperature is subtracted by MH_DUTY_CYC_RATE. If the subtraction result is greater than zero, the corresponding MEM_HOT# is asserted; otherwise, the MEM_HOT# is de-asserted. By setting this field to zero, the MEM_HOT becomes level mode.
7:0	RW	0x1f	MH_BASE_TEMP (mh_base_temp): MEM_HOT base temperature. The base temp is subtracted from the hottest DIMM temp to obtain a relative temperature substituted to zero if negative. The delta temperature is used to generate the MEM_HOT#.

3.1.8 mh_sense_500ns_cfg

MEMHOT Sense and 500 ns Config.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x10c**				

Bit	Attr	Default	Description
25:16	RW	0xc8	MH_SENSE_PERIOD (mh_sense_period): MEMHOT Input Sense Period in number of CNTR_500_Nanoseconds BIOS calculate number of CNTR_500_NANOSEC for 50 microseconds/100 microseconds/200 microseconds/400 microseconds
15:13	RW	0x2	MH_IN_SENSE_ASSERT (mh_in_sense_assert): MEMHOT Input Sense Assertion Time in number of CNTR_500_NANOSEC. BIOS calculate number of CNFG_500_NANOSEC for 1 microseconds/2 microseconds inputsense duration Here is MH_IN_SENSE_ASSERT ranges: 0 or 1 Reserved 2 - 7 1 microseconds - 3.5 microseconds sense assertion time in 500nsec increment

9:0	RWS	0x190	CNFG_500_NANOSEC (cnfg_500_nanosec):
			500ns equivalent in DCLK. BIOS calculate number of DCLK to be equivalent to 500 nanoseconds. This value is loaded into CNTR_500_NANOSEC when it is decremented to zero. For pre-Si validation, minimum 2 can be set to speed up the simulation.
			The following are the recommended CNFG_500_NANOSEC values based from each DCLK frequency:
			DCLK = 400 MHz, CNFG_500_NANOSEC = 0C8h
			DCLK = 533 MHz, CNFG_500_NANOSEC = 10Ah
			DCLK = 667 MHz, CNFG_500_NANOSEC = 14Dh
			DCLK = 800 MHz, CNFG_500_NANOSEC = 190h
			DCLK = 933 MHz, CNFG_500_NANOSEC = 1D2h

3.1.9 mh_dtycyc_min_asrt_cntr_[0:1]

MEMHOT Duty Cycle Period and minimum Assertion Counter.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Bus:	**1**	**Device:**	**29**	**Function:**	**0**
Offset:	**0x110, 0x114**				

Bit	Attr	Default	Description
31:20	RO-V	0x0	MH_MIN_ASRTN_CNTR (mh_min_asrtn_cntr): MEM_HOT[1:0]# Minimum Assertion Time Current Count in number of CNTR_500_NANOSEC decrement by 1 every CNTR_500_NANOSEC. When the counter is zero, the counter is remain at zero and it is only loaded with MH_MIN_ASRTN only when MH_DUTY_CYC_PRD_CNTR is reloaded.
19:0	RW_LV	0x0	MH_DUTY_CYC_PRD_CNTR (mh_duty_cyc_prd_cntr): MEM_HOT[1:0]# DUTY Cycle Period Current Count in number of CNTR_500_NANOSEC decrement by 1 every CNTR_500_NANOSEC. When the counter is zero, the next cycle is loaded with MH_DUTY_CYC_PRD. PMSI pause (at quiencense) and resume (at wipe).

3.1.10 mh_io_500ns_cntr

MEMHOT Input Output and 500ns Counter.

Type: **CFG** **Bus:** **1** **Offset:** **0x118**		**PortID:** **N/A** **Device:** **15**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
31:22	RW_LV	0x0	MH1_IO_CNTR (mh1_io_cntr): MEM_HOT[1:0]# Input Output Counter in number of CNTR_500_NANOSEC. When MH0_IO_CNTR is zero, the counter is loaded with MH_SENSE_PERIOD in the next CNTR_500_NANOSEC. When count is greater than MH_IN_SENSE_ASSERT, the MEM_HOT1# output driver may be turn on if the corresponding MEM_HOT#event is asserted. The receiver is turned off during this time. When count is equal or less than MH_IN_SENSE_ASSERT, MEM_HOT[1:0]# output is disabled and receiver is turned on. Hardware will decrement this counter by 1 every time CNTR_500_NANOSEC is decremented to zero. When the counter is zero, the next CNFG_500_NANOSEC count is loaded with MH_IN_SENSE_ASSERT. This counter is subject to PMSI pause (at quiencense) and resume (at wipe).
21:12	RW_LV	0x0	MH0_IO_CNTR (mh0_io_cntr): MEM_HOT[1:0]# Input Output Counter in number of CNTR_500_NANOSEC. When MH_IO_CNTR is zero, the counter is loaded with MH_SENSE_PERIOD in the next CNTR_500_NANOSEC. When count is greater than MH_IN_SENSE_ASSERT, the MEM_HOT[1:0]# output driver may be turn on if the corresponding MEM_HOT#event is asserted. The receiver is turned off during this time. When count is equal or less than MH_IN_SENSE_ASSERT, MEM_HOT[1:0]# output is disabled and receiver is turned on. BIOS calculate number of CNTR_500_NANOSEC hardware will decrement this register by 1 every CNTR_500_NANOSEC. When the counter is zero, the next CNTR_500_NANOSEC count is loaded with MH_IN_SENSE_ASSERT. This counter is subject to PMSI pause (at quiencense) and resume (at wipe).
9:0	RW_LV	0x0	CNTR_500_NANOSEC (cntr_500_nanosec): 500 ns base counters used for the MEMHOT counters and the SMBus counters. BIOS calculate number of DCLK to be equivalent to 500 nanoseconds. CNTR_500_NANOSEC hardware will decrement this register by 1 every CNTR_500_NANOSEC. When the counter is zero, the next CNTR_500_NANOSEC count is loaded with CNFG_500_NANOSEC. This counter is subject to PMSI pause (at quiencense) and resume (at wipe).

3.1.11 mh_chn_astn

MEMHOT Domain Channel Association

Type: CFG **Bus:** 1 **Offset:** 0x11c		**PortID:** N/A **Device:** 15	**Function:** 0

Bit	Attr	Default	Description
23:20	RO	0xb	MH1_2ND_CHN_ASTN (mh1_2nd_chn_astn): MemHot[1]# 2nd Channel Association bit 23: is valid bit. **Note:** Valid bit means the association is valid and it does not implies the channel is populated. Bit 22-20: 2nd channel ID within this MEMHOT domain. **Note:** This register is hardcoded in design. It is read-accessible by firmware. Design must make sure this register is not removed by downstream tools.
19:16	RO	0xa	MH1_1ST_CHN_ASTN (mh1_1st_chn_astn): MemHot[1]# 1st Channel Association bit 19: is valid bit. **Note:** Valid bit means the association is valid and it does not implies the channel is populated. bit 18-16: 1st channel ID within this MEMHOT domain **Note:** This register is hardcoded in design. It is read-accessible by firmware. Design must make sure this register is not removed by downstream tools.
7:4	RO	0x9	MH0_2ND_CHN_ASTN (mh0_2nd_chn_astn): MemHot[0]# 2nd Channel Association bit 7: is valid bit. **Note**: Valid bit means the association is valid and it does not implies the channel is populated. bit 6-4: 2nd channel ID within this MEMHOT domain **Note:** This register is hardcoded in design. It is read-accessible by firmware. Design must make sure this register is not removed by downstream tools.
3:0	RO	0x8	MH0_1ST_CHN_ASTN (mh0_1st_chn_astn): MemHot[0]# 1st Channel Association bit 3: is valid bit. **Note:** Valid bit means the association is valid and it does not implies the channel is populated or exist. bit 2-0: 1st channel ID within this MEMHOT domain **Note:** This register is hardcoded in design. It is read-accessible by firmware. Design must make sure this register is not removed by downstream tools.

3.1.12 mh_temp_stat

MEMHOT TEMP Status.

Type: CFG **Bus:** 1 **Offset:** 0x120	**PortID:** N/A **Device:** 15	**Function:** 0	
Bit	**Attr**	**Default**	**Description**
31:31	RW-V	0x0	MH1_DIMM_VAL (mh1_dimm_val): Valid if set. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID and set the valid bit. MEMHOT hardware logic process the corresponding MEMHOT data when there is a MEMHOT event. Upon processing, the valid bit is reset. PCODE can write over existing valid temperature since a valid temperature may not occur during a MEMHOT event. If PCODE set the valid bit occur at the same cycle that the MEMHOT logic processing and try to clear, the PCODE set will dominate since it is a new temperature is updated while processing logic tries to clear an existing temperature.
30:28	RW	0x0	MH1_DIMM_CID (mh1_dimm_cid): Hottest DIMM Channel ID for MEM_HOT[1]#. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID.
27:24	RW	0x0	MH1_DIMM_ID (mh1_dimm_id): Hottest DIMM ID for MEM_HOT[1]#. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID.
23:16	RW	0x0	MH1_TEMP (mh1_temp): Hottest DIMM Sensor Reading for MEM_HOT[1]# - This reading represents the temperature of the hottest DIMM. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID. **Note**: iMC hardware load this value into the MEMHOT duty cycle generator counter since PCODE may update this field at different rate/time. This field is ranged from 0 to 127, that is, the most significant bit is always zero.
15:15	RW-V	0x0	MH0_DIMM_VAL (mh0_dimm_val): Valid if set. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID and set the valid bit. MEMHOT hardware logic process the corresponding MEMHOT data when there is a MEMHOT event. Upon processing, the valid bit is reset. PCODE can write over existing valid temperature since a valid temperature may not occur during a MEMHOT event. If PCODE set the valid bit occur at the same cycle that the MEMHOT logic processing and try to clear, the PCODE set will dominate since it is a new temperature is updated while processing logic tries to clear an existing temperature.
14:12	RW	0x0	MH0_DIMM_CID (mh0_dimm_cid): Hottest DIMM Channel ID for MEM_HOT[0]#. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID.
11:8	RW	0x0	MH0_DIMM_ID (mh0_dimm_id): Hottest DIMM ID for MEM_HOT[0]#. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID.
7:0	RW	0x0	MH0_TEMP (mh0_temp): Hottest DIMM Sensor Reading for MEM_HOT[0]# - This reading represents the temperature of the hottest DIMM. PCODE search the hottest DIMM temperature and write the hottest temperature and the corresponding Hottest DIMM CID/ID. **Note**: iMC hardware load this value into the MEMHOT duty cycle generator counter since PCODE may update this field at different rate/time. This field is ranged from 0 to 127, that is, the most significant bit is always zero.

3.1.13 mh_ext_stat

Capture externally asserted MEM_HOT[1:0]# assertion detection.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x124**				

Bit	Attr	Default	Description
1:1	RW1C	0x0	MH_EXT_STAT_1 (mh_ext_stat_1): MEM_HOT[1]# assertion status at this sense period. Set if MEM_HOT[1]# is asserted externally for this sense period, this running status bit will automatically updated with the next sensed value in the next MEMHOT input sense phase.
0:0	RW1C	0x0	MH_EXT_STAT_0 (mh_ext_stat_0): MEM_HOT[0]# assertion status at this sense period. Set if MEM_HOT[0]# is asserted externally for this sense period, this running status bit will automatically updated with the next sensed value in the next MEMHOT input sense phase.

3.1.14 smb_stat_[0:1]

SMBus Status

This register provides the interface to the SMBus/I^2C* SCL and SDA signals that is used to access the Serial Presence Detect EEPROM (SPD) or Thermal Sensor on DIMM (TSOD) that defines the technology, configuration, and speed of the DIMMs controlled by iMC.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x180, 0x190**				

Bit	Attr	Default	Description
31:31	RO-V	0x0	SMB_RDO (smb_rdo): Read Data Valid This bit is set by iMC when the Data field of this register receives read data from the SPD/TSOD after completion of an SMBus read command. It is cleared by iMC when a subsequent SMBus read command is issued.
30:30	RO-V	0x0	SMB_WOD (smb_wod): Write Operation Done This bit is set by iMC when a SMBus Write command has been completed on the SMBus. It is cleared by iMC when a subsequent SMBus Write command is issued.
29:29	RO-V	0x0	SMB_SBE (smb_sbe): SMBus Error This bit is set by iMC if an SMBus transaction (including the TSOD polling or message channel initiated SMBus access) that does not complete successfully (non-Ack has been received from slave at expected Ack slot of the transfer). If a slave device is asserting clock stretching, IMC does not have logic to detect this condition to set the SBE bit directly; however, the SMBus master will detect the error at the corresponding transaction's expected ACK slot. Once SMBUS_SBE bit is set, iMC stops issuing hardware initiated TSOD polling SMBUS transactions until the SMB_SBE is cleared. iMC will not increment the SMB_STAT_x.TSOD_SA until the SMB_SBE is cleared. Manual SMBus command interface is not affected, that is, new command issue will clear the SMB_SBE like A0 silicon behavior.

Type:	CFG	PortID:	N/A		
Bus:	1	Device:	15	Function:	0
Offset:	0x180, 0x190				

Bit	Attr	Default	Description
28:28	ROS_V	0x0	SMB_BUSY (smb_busy): SMBus Busy state. This bit is set by iMC while an SMBus/I2C command (including TSOD command issued from IMC hardware) is executing. Any transaction that is completed normally or gracefully will clear this bit automatically. By setting the SMB_SOFT_RST will also clear this bit. This register bit is sticky across reset so any surprise reset during pending SMBus operation will sustain the bit assertion across surprised warm-reset. BIOS reset handler can read this bit before issuing any SMBus transaction to determine whether a slave device may need special care to force the slave to idle state (for example, by means of the clock override toggling SMB_CKOVRD and/or by means of induced time-out by asserting SMB_CKOVRD for 25-35 ms).
27:24	RO-V	0x7	Last Issued TSOD Slave Address (tsod_sa): This field captures the last issued TSOD slave address. Here is the slave address and the DDR CHN and DIMM slot mapping: Slave Address: 0 -- Channel: Even Chn; Slot #: 0 Slave Address: 1 -- Channel: Even Chn; Slot #: 1 Slave Address: 2 -- Channel: Even Chn; Slot #: 2 Slave Address: 3 -- Channel: Even Chn; Slot #: 3 (reserved) Slave Address: 4 -- Channel: Odd Chn; Slot #: 0 Slave Address: 5 -- Channel: Odd Chn; Slot #: 1 Slave Address: 6 -- Channel: Odd Chn; Slot #: 2 Slave Address: 7 -- Channel: Odd Chn; Slot #: 3 (reserved) A value of 8 in this register indicates to poll MXB temperature rather than a DIMM temperature, values above 0x8 are invalid. Since this field only captures the TSOD polling slave address. During SMB error handling, software should check the hung SMB_TSOD_POLL_EN state before disabling the SMB_TSOD_POLL_EN in order to qualify whether this field is valid.
15:0	RO-V	0x0	SMB_RDATA (smb_rdata): Read DataHolds data read from SMBus Read commands. Since TSOD/EEPROM are I2C* devices and the byte order is MSByte first in a word read, reading of I2C using word read should return SMB_RDATA[15:8] = I2C_MSB and SMB_RDATA[7:0] = I2C_LSB. If reading of I2C using byte read, the SMB_RDATA[15:8] = dont care; SMB_RDATA[7:0] = readbyte. If we have a SMB slave connected on the bus, reading of the SMBus slave using word read should return SMB_RDATA[15:8] = SMB_LSB and SMB_RDATA[7:0] = SMB_MSB. If the software is not sure whether the target is I2C or SMBus slave, use byte access.

3.1.15 smbcmd_[0:1]

A write to this register initiates a DIMM EEPROM access through the SMBus/I^{2}C.

Type: CFG **Bus:** 1 **Offset:** 0x184, 0x194	**PortID:** N/A **Device:** 15	**Function:** 0	
Bit	**Attr**	**Default**	**Description**
31:31	RW-V	0x0	SMB_CMD_TRIGGER (smb_cmd_trigger): CMD trigger: After setting this bit to 1, the SMBus master will issue the SMBus command using the other fields written in SMBCMD_[0:1] and SMBCntl_[0:1]. **Note**: The '-V' in the attribute implies the hardware will reset this bit when the SMBus command is being started.
30:30	RWS	0x0	SMB_PNTR_SEL (smb_pntr_sel): Pointer Selection: SMBus/I2C present pointer based access enable when set; otherwise, use random access protocol. Hardware based TSOD polling will also use this bit to enable the pointer word read. **Important Note:** Processor hardware based TSOD polling can be configured with pointer based access. If software manually issue SMBus transaction to other address, that is changing the pointer in the slave device, it is software's responsibility to restore the pointer in each TSOD before returning to hardware based TSOD polling while keeping the SMB_PNTR_SEL = 1.
29:29	RWS	0x0	SMB_WORD_ACCESS (smb_word_access): word access: SMBus/I2C word 2B access when set; otherwise, it is a byte access.
28:28	RWS	0x0	SMB_WRT_PNTR (smb_wrt_pntr): Bit[28:27] = 00: SMBus Read Bit[28:27] = 01: SMBus Write Bit[28:27] = 10: illegal combination Bit[28:27] = 11: Write to pointer register SMBus/I2C pointer update (byte). bit 30, and 29 are ignored. **Note**: SMBCntl_[0:1] [26] will NOT disable WrtPntr update command.
27:27	RWS	0x0	SMB_WRT_CMD (smb_wrt_cmd): When '0', it's a read command When '1', it's a write command
26:24	RWS	0x0	SMB_SA (smb_sa): Slave Address: This field identifies the DIMM SPD/TSOD to be accessed.
23:16	RWS	0x0	SMB_BA (smb_ba): Bus Txn Address: This field identifies the bus transaction address to be accessed. **Note**: In WORD access, 23:16 specifies 2B access address. In Byte access, 23:16 specified 1B access address.
15:0	RWS	0x0	SMB_WDATA (smb_wdata): Write Data: Holds data to be written by SPDW commands. Since TSOD/EEPROM are I2C devices and the byte order is MSByte first in a word write, writing of I2C using word write should use SMB_WDATA[15:8] = I2C_MSB and SMB_WDATA[7:0] = I2C_LSB. If writing of I2C using byte write, the SMB_WDATA[15:8] = dont care; SMB_WDATA[7:0] = writebyte. If we have a SMB slave connected on the bus, writing of the SMBus slave using word write should use SMB_WDATA[15:8] = SMB_LSB and SMB_WDATA[7:0] = SMB_MSB. It is software responsibility to determine the byte order of the slave access.

3.1.16 smbcntl_[0:1]

SMBus Control

Type: CFG **Bus:** 1 **Offset:** 0x188, 0x198	**PortID:** N/A **Device:** 15	**Function:** 0

Bit	Attr	Default	Description
31:28	RWS	0xa	SMB_DTI (smb_dti): Device Type Identifier: This field specifies the device type identifier. Only devices with this device-type will respond to commands. '0011' specifies TSOD. '1010' specifies EEPROM's. '0110' specifies a write-protect operation for an EEPROM. Other identifiers can be specified to target non-EEPROM devices on the SMBus. **Note:** IMC based hardware TSOD polling uses hardcoded DTI. Changing this field has no effect on the hardware based TSOD polling.
27:27	RWS_V	0x1	SMB_CKOVRD (smb_ckovrd): Clock Override '0' Clock signal is driven low, overriding writing a '1' to CMD. '1' Clock signal is released high, allowing normal operation of CMD. Toggling this bit can be used to 'budge' the port out of a 'stuck' state. Software can write this bit to 0 and the SMB_SOFT_RST to 1 to force hung SMBus controller and the SMB slaves to idle state without using power good reset or warm reset. **Note:** Software needs to set the SMB_CKOVRD back to 1 after 35 ms in order to force slave devices to time-out in case there is any pending transaction. The corresponding SMB_STAT_x.SMB_SBE error status bit may be set if there was such pending transaction time-out (non-graceful termination). If the pending transaction was a write operation, the slave device content may be corrupted by this clock override operation. A subsequent SMB command will automatically cleared the SMB_SBE. iMC added SMBus time-out control timer in B0. When the time-out control timer expired, the SMBCKOVRD# will "de-assert", that is return to 1 value and clear the SMBSBE0.
26:26	RW-LB	0x1	SMB_DIS_WRT (smb_dis_wrt): Disable SMBus Write Writing a '0' to this bit enables CMD to be set to 1; Writing a 1 to force CMD bit to be always 0, that is disabling SMBus write. This bit can only be written in SMMode. SMBus Read is not affected. I2C Write Pointer Update Command is not affected. **Important Note:** Since BIOS is the source to update SMBCNTL_x register initially after reset, it is important to determine whether the SMBus can have write capability before writing any upper bits (bits 24-31) by means of the byte-enable configuration write (or writing any bit within this register by means of 32b config write) within the SMBCNTL register.
23:23	RW	0x0	smb_sbe_en: SMBus error recovery enable if set; otherwise, A0 behavior.
22:22	RW	0x0	smb_sbe_smi_en: Enable SMI generation when SMB_SBE is 0 -- 1.
21:21	RW	0x0	smb_sbe_err0_en: Enable ERR0 assertion when SMB_SBE is 0 -- 1.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x188, 0x198**				

Bit	Attr	Default	Description
10:10	RW	0x0	SMB_SOFT_RST (smb_soft_rst): SMBus software reset strobe to graceful terminate pending transaction after ACK and keep the SMB from issuing any transaction until this bit is cleared. If slave device is hung, software can write this bit to 1 and the SMB_CKOVRD to 0 (for more than 35ms)to force hung the SMB slaves to time-out and put it in idle state without using power good reset or warm reset. **Note**: Software needs to set the SMB_CKOVRD back to 1 after 35 ms in order to force slave devices to time-out in case there is any pending transaction. The corresponding SMB_STAT_x.SMB_SBE error status bit may be set if there was such pending transaction time-out (non-graceful termination). If the pending transaction was a write operation, the slave device content may be corrupted by this clock override operation. A subsequent SMB command will automatically cleared the SMB_SBE. If the IMC hardware perform SMB time-out with the SMB_SBE_EN = 1. Software should simply clear the SMB_SBE and SMB_SOFT_RST sequentially after writing the SMB_CKOVRD = 0 and SMB_SOFT_RST = 1 asserting clock override and perform graceful txn termination. Hardware will automatically de-assert the SMB_CKOVRD update to 1 after the pre-configured 35 ms/65 ms timeout.
8:8	RW-LB	0x0	SMB_TSOD_POLL_EN (smb_tsod_poll_en): TSOD polling enable '0': disable TSOD polling and enable SPDCMD accesses. '1': disable SPDCMD access and enable TSOD polling. It is important to make sure no pending SMBus transaction and the TSOD polling must be disabled (and pending TSOD polling must be drained) before changing the TSOD_POLL_EN.
7:0	RW-LB	0x0	TSOD_PRESENT for the lower and upper channels (tsod_present): DIMM slot mask to indicate whether the DIMM is equipped with TSOD sensor. Bit 7: must be programmed to zero. Upper channel slot #3 is not supported Bit 6: TSOD PRESENT at upper channel (ch 1 or ch 3) slot #2 Bit 5: TSOD PRESENT at upper channel (ch 1 or ch 3) slot #1 Bit 4: TSOD PRESENT at upper channel (ch 1 or ch 3) slot #0 Bit 3: must be programmed to zero. Lower channel slot #3 is not supported Bit 2: TSOD PRESENT at lower channel (ch 0 or ch 2) slot #2 Bit 1: TSOD PRESENT at lower channel (ch 0 or ch 2) slot #1 Bit 0: TSOD PRESENT at lower channel (ch 0 or ch 2) slot #0

3.1.17 smb_tsod_poll_rate_cntr_[0:1]

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x18c, 0x19c**				

Bit	Attr	Default	Description
17:0	RW_LV	0x0	SMB_TSOD_POLL_RATE_CNTR (smb_tsod_poll_rate_cntr): TSOD poll rate counter. When it is decremented to zero, reset to zero or written to zero, SMB_TSOD_POLL_RATE value is loaded into this counter and appear the updated value in the next DCLK.

3.1.18 smb_period_cfg

SMBus Clock Period Configuration

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x1a0**				

Bit	Attr	Default	Description
31:16	RWS	0x445c	smb_tlow_timeout: Upper 16b of the 18b SMBus Time-Out Timer Configuration in unit of MH_SENSE_500NS_CFG.CNFG_500_NANOSEC. The lower 2b of the 18b counter config is always 00. Assuming the CNFG_500_NANOSEC is set at 500 ns: For 35 ms time out, configure this register to 445°C For 65 ms time out, configure this register to 7EF4
15:0	RWS	0xfa0	SMB_CLK_PRD (smb_clk_prd): This field specifies both SMBus Clock in number of DCLK. **Note:** In order to generate a 50% duty cycle SCL, half of the SMB_CLK_PRD is used to generate SCL high. SCL must stay low for at least another half of the SMB_CLK_PRD before pulling high. It is recommend to program an even value in this field since the hardware is simply doing a right shift for the divided by 2 operation. For pre-Si validation, minimum 8 can be set to speed up the simulation. **Note:** The 100 KHz SMB_CLK_PRD default value is calculated based on 800 MTs (400 MHz) DCLK.

3.1.19 smb_period_cntr

SMBus Clock Period Counter

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x1a4**				

Bit	Attr	Default	Description
31:16	RO-V	0x0	SMB1_CLK_PRD_CNTR (smb1_clk_prd_cntr): SMBus #1 Clock Period Counter for Ch 23. This field is the current SMBus Clock Period Counter Value.
15:0	RO-V	0x0	SMB0_CLK_PRD_CNTR (smb0_clk_prd_cntr): SMBus #0 Clock Period Counter for Ch 01. This field is the current SMBus Clock Period Counter Value.

3.1.20 smb_tsod_poll_rate

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**0**
Offset:	**0x1a8**				

Bit	Attr	Default	Description
17:0	RWS	0x3e800	SMB_TSOD_POLL_RATE (smb_tsod_poll_rate): TSOD poll rate configuration between consecutive TSOD accesses to the TSOD devices on the same SMBus segment. This field specifies the TSOD poll rate in number of 500 ns per CNFG_500_NANOSEC register field definition.

3.2 Device 15 Functions 2-5

Table 3-2. IMC Device 15 Function 2–5 Register Address Map

Register Name	Offset	Size
pxpcap	0x40	32
dimmmtr_0	0x80	32
dimmmtr_1	0x84	32
dimmmtr_2	0x88	32
pxpenhcap	0x100	32

3.2.1 pxpcap

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**2,3,4,5**
Offset:	**0x40**				

Bit	Attr	Default	Description
29:25	RO	0x0	Interrupt Message Number (interrupt_message_number): N/A for this device
24:24	RO	0x0	Slot Implemented (slot_implemented): N/A for integrated endpoints
23:20	RO	0x9	Device/Port Type (device_port_type): Device type is Root Complex Integrated Endpoint
19:16	RO	0x1	Capability Version (capability_version): PCI Express* Capability is Compliant with Version 1.0 of the PCI Express* Specification. **Note:** This capability structure is not compliant with Versions beyond 1.0, since they require additional capability registers to be reserved. The only purpose for this capability structure is to make enhanced configuration space available. Minimizing the size of this structure is accomplished by reporting version 1.0 compliancy and reporting that this is an integrated root port device. As such, only three Dwords of configuration space are required for this structure.
15:8	RO	0x0	Next Capability Pointer (next_ptr): Pointer to the next capability. Set to 0 to indicate there are no more capability structures.
7:0	RO	0x10	Capability ID (capability_id): Provides the PCI Express* capability ID assigned by PCI-SIG.

3.2.2 dimmmtr_[0:2]

DIMM Memory Technology.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**15**	**Function:**	**2,3,4,5**
Offset:	**0x80, 0x84, 0x88**				

Bit	Attr	Default	Description
20:20	RW-LB	0x0	rsvd:
19:16	RW-LB	0x0	RANK_DISABLE control (rank_disable): RANK Disable Control to disable patrol, refresh and ZQCAL operation. This bit setting must be set consistently with TERM_RNK_MSK, that is both corresponding bits cannot be set at the same time. In the other word, a disabled rank must not be selected for the termination rank. RANK_DISABLE[3], that is bit 19: rank 3 disable. **Note:** DIMMMTR_2.RANKDISABLE[3] is don't care since DIMM 2 must not be quad-rank RANK_DISABLE[2], that is bit 18: rank 2 disable. **Note:** DIMMMTR_2.RANKDISABLE[2] is don't care since DIMM 2 must not be quad-rank RANK_DISABLE[1], that is bit 17: rank 1 disable RANK_DISABLE[0], that is bit 16: rank 0 disable when set, no patrol or refresh will be perform on this rank. ODT termination is not affected by this bit.
14:14	RW-LB	0x0	DIMM_POP (dimm_pop): DIMM populated if set; otherwise, unpopulated.
13:12	RW-LB	0x0	RANK_CNT (rank_cnt): 00 - SR 01 - DR 10 - QR 11 - reserved
8:7	RW-LB	0x0	DDR3_WIDTH (ddr3_width): 00 - x4 01 - x8 10 - x16 11 - reserved Used to determine if a configuration is capable of supporting DDDC.
6:5	RW-LB	0x0	DDR3_DNSTY (ddr3_dnsty): 00 - 1 Gb 01 - 2 Gb 10 - 4 Gb 11 - 8 Gb
4:2	RW-LB	0x0	RA_WIDTH (ra_width): 000 - reserved - 13 bits 010 - 14 bits 011 - 15 bits 100 - 16 bits 101 - 17 bits HDRL, if DISABLE_EXTENDED_ADDR_DIMM is 1, setting 101 is decoded as 100. (Such configuration is not supported) 110 - 18 bits HDRL, if DISABLE_EXTENDED_ADDR_DIMM is 1, setting 110 is decoded as 100. (Such configuration is not supported) 111: reserved
1:0	RW-LB	0x0	CA_WIDTH (ca_width): 00 - 10 bits 01 - 11 bits 10 - 12 bits 11 - reserved

3.2.3 pxpenhcap

This field points to the next Capability in extended configuration space.

<table>
<tr><td colspan="4"><b>Type:</b> CFG   <b>PortID:</b> N/A
<b>Bus:</b> 1   <b>Device:</b> 15   <b>Function:</b> 2,3,4,5
<b>Offset:</b> 0x100</td></tr>
<tr><th>Bit</th><th>Attr</th><th>Default</th><th>Description</th></tr>
<tr><td>31:20</td><td>RO</td><td>0x0</td><td>Next Capability Offset (next_capability_offset):</td></tr>
<tr><td>19:16</td><td>RO</td><td>0x0</td><td>Capability Version (capability_version):
Indicates there are no capability structures in the enhanced configuration space.</td></tr>
<tr><td>15:0</td><td>RO</td><td>0x0</td><td>Capability ID (capability_id):
Indicates there are no capability structures in the enhanced configuration space.</td></tr>
</table>

3.3 Device 16 Functions 0, 1, 4, 5

Table 3-3. IMC Device 16 Function 0, 1, 4, 5 Register Address Map

Register Name	Offset	Size
pxpcap	0x40	32
chn_temp_cfg	0x108	32
chn_temp_stat	0x10c	32
dimm_temp_oem_0	0x110	32
dimm_temp_oem_1	0x114	32
dimm_temp_oem_2	0x118	32
dimm_temp_th_0	0x120	32
dimm_temp_th_1	0x124	32
dimm_temp_th_2	0x128	32
dimm_temp_thrt_lmt_0	0x130	32
dimm_temp_thrt_lmt_1	0x134	32
dimm_temp_thrt_lmt_2	0x138	32
dimm_temp_ev_ofst_0	0x140	32
dimm_temp_ev_ofst_1	0x144	32
dimm_temp_ev_ofst_2	0x148	32
dimmtempstat_0	0x150	32
dimmtempstat_1	0x154	32
dimmtempstat_2	0x158	32
thrt_pwr_dimm_0	0x190	16
thrt_pwr_dimm_1	0x192	16
thrt_pwr_dimm_2	0x194	16
tcdbp	0x200	32
tcrap	0x204	32
tcrwp	0x208	32
tcothp	0x20c	32
tcrfp	0x210	32
tcrftp	0x214	32
tcsrftp	0x218	32
tcmr2shadow	0x21c	32
tczqcal	0x220	32
tcstagger_ref	0x224	32
tcmr0shadow	0x22c	32
rpqage	0x234	32
idletime	0x238	32
rdimmtimingcntl	0x23c	32
rdimmtimingcntl2	0x240	32
tcmrs	0x244	32
mc_init_stat_c	0x280	32

3.3.1 pxpcap

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x40**				

Bit	Attr	Default	Description
7:0	RO	0x10	Capability ID (capability_id): Provides the PCI Express* capability ID assigned by PCI-SIG.

3.3.2 chn_temp_cfg

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x108**				

Bit	Attr	Default	Description
31:31	RW	0x1	OLTT_EN (oltt_en): Enable OLTT temperature tracking
29:29	RW	0x0	CLTT_OR_PCODE_TEMP_MUX_SEL (cltt_or_pcode_temp_mux_sel): The TEMP_STAT byte update multiplex select control to direct the source to update DIMMTEMPSTAT_[0:3][7:0]:0: Corresponding to the DIMM TEMP_STAT byte from PCODE_TEMP_OUTPUT. 1: TSOD temperature reading from CLTT logic.
28:28	RW-O	0x1	CLTT_DEBUG_DISABLE_LOCK (cltt_debug_disable_lock): lock bit of DIMMTEMPSTAT_[0:3][7:0]:Set this lock bit to disable configuration write to DIMMTEMPSTAT_[0:3][7:0]. When this bit is clear, system debug test software can update the DIMMTEMPSTAT_[0:3][7:0] to verify various temperature sceneries.
27:27	RW	0x1	Enables thermal bandwidth throttling limit (bw_limit_thrt_en):
23:16	RW	0x0	THRT_EXT (thrt_ext): Maximum number of throttled transactions to be issued during BWLIMITTF due to externally asserted MEMHOT#.
15:15	RW	0x0	THRT_ALLOW_ISOCH (thrt_allow_isoch): When this bit is zero, MC will lower CKE during Thermal Throttling, and ISOCH is blocked. When this bit is one, MC will NOT lower CKE during Thermal Throttling, and ISOCH will be allowed base on bandwidth throttling setting. However, setting this bit would mean more power consumption due to CKE is asserted during thermal or power throttling. This bit can be updated dynamically in independent channel configuration only. For lock-step configuration, this bit must be statically set during IOSAV mode before enabling the lock-step operation. Dynamic update in lock-step mode will put the two lock-stepped channels out-of-sync and cause functional failure or silent data corruption.
10:0	RW	0x3ff	BW_LIMIT_TF (bw_limit_tf): BW Throttle Window Size in DCLK

3.3.3 chn_temp_stat

Type: CFG **Bus:** 1 **Offset:** 0x10c	**PortID:** N/A **Device:** 16	**Function:** 0,1,4,5	

Bit	Attr	Default	Description
3:3	RW1C	0x0	Event Asserted MXB (ev_asrt_mxb): Event Asserted on MXB
2:2	RW1C	0x0	Event Asserted on DIMM ID 2 (ev_asrt_dimm2): Event Asserted on DIMM ID 2
1:1	RW1C	0x0	Event Asserted on DIMM ID 1 (ev_asrt_dimm1): Event Asserted on DIMM ID 1
0:0	RW1C	0x0	Event Asserted on DIMM ID 0 (ev_asrt_dimm0): Event Asserted on DIMM ID 0

3.3.4 dimm_temp_oem_[0:2]

Type: CFG **Bus:** 1 **Offset:** 0x110, 0x114, 0x118	**PortID:** N/A **Device:** 16	**Function:** 0,1,4,5	

Bit	Attr	Default	Description
26:24	RW	0x0	TEMP_OEM_HI_HYST (temp_oem_hi_hyst): Positive going Threshold Hysteresis Value. This value is subtracted from TEMPOEMHI to determine the point where the asserted status for that threshold will clear. Set to 00h if sensor does not support positive-going threshold hysteresis
18:16	RW	0x0	TEMP_OEM_LO_HYST (temp_oem_lo_hyst): Negative going Threshold Hysteresis Value. This value is added to TEMPOEMLO to determine the point where the asserted status for that threshold will clear. Set to 00h if sensor does not support negative-going threshold hysteresis.
15:8	RW	0x50	TEMP_OEM_HI (temp_oem_hi): Upper Threshold value - T_{CASE} threshold at which to Initiate System Interrupt (SMI or MEMHOT#) at a+ going rate. **Note**: The default value is listed in decimal.valid range: 32 - 127 in degree (C). Others: reserved.
7:0	RW	0x4b	TEMP_OEM_LO (temp_oem_lo): Lower Threshold Value - T_{CASE} threshold at which to Initiate System Interrupt (SMI or MEMHOT#) at a - going rate. **Note**: The default value is listed in decimal.valid range: 32 - 127 in degree (C). Others: reserved

3.3.5 dimm_temp_th_[0:2]

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x120, 0x124, 0x128**				

Bit	Attr	Default	Description
26:24	RW	0x0	TEMP_THRT_HYST (temp_thrt_hyst): Positive going Threshold Hysteresis Value. Set to 00h if sensor does not support positive-going threshold hysteresis. This value is subtracted from TEMP_THRT_XX to determine the point where the asserted status for that threshold will clear.
23:16	RW	0x5f	TEMP_HI (temp_hi): T_{CASE} threshold at which to Initiate THRTCRIT and assert THERMTRIP# valid range: 32 - 127 in degree (C). **Note**: The default value is listed in decimal. FF: Disabled Others: reserved. TEMP_HI should be programmed so it is greater than TEMP_MID
15:8	RW	0x5a	TEMP_MID (temp_mid): T_{CASE} threshold at which to Initiate THRTHI and assert valid range: 32 - 127 in degree (C). **Note:** The default value is listed in decimal. FF: Disabled Others: reserved. TEMP_MID should be programmed so it is less than TEMP_HI
7:0	RW	0x55	TEMP_LO (temp_lo): T_{CASE} threshold at which to Initiate 2x refresh andor THRTMID and initiate Interrupt (MEMHOT#). **Note:** The default value is listed in decimal.valid range: 32 - 127 in degree (C). FF: Disabled Others: reserved. TEMP_LO should be programmed so it is less than TEMP_MID

3.3.6 dimm_temp_thrt_lmt_[0:2]

All three THRT_CRIT, THRT_HI and THRT_MID are per DIMM BW limit, that is all activities (ACT, READ, WRITE) from all ranks within a DIMM are tracked together in one DIMM activity counter.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x130, 0x134, 0x138**				

Bit	Attr	Default	Description
23:16	RW	0x0	THRT_CRIT (thrt_crit): Maximum number of throttled transactions (ACT, READ, WRITE) to be issued during BWLIMITTF.
15:8	RW	0xf	THRT_HI (thrt_hi): Maximum number of throttled transactions (ACT, READ, WRITE) to be issued during BWLIMITTF.
7:0	RW	0xff	THRT_MID (thrt_mid): Maximum number of throttled transactions (ACT, READ, WRITE) to be issued during BWLIMITTF.

3.3.7 dimm_temp_ev_ofst_[0:2]

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x140, 0x144, 0x148**				

Bit	Attr	Default	Description
31:24	RO	0x0	TEMP_AVG_INTRVL (temp_avg_intrvl): Temperature data is averaged over this period. At the end of averaging period (ms), averaging process starts again. 0x1 - 0xFF Averaging data is read by means of TEMPDIMM STATUSREGISTER (Byte 1/2) as well as used for generating hysteresis based interrupts. 00 Instantaneous Data (non-averaged) is read by means of TEMPDIMM STATUSREGISTER (Byte 1/2) as well as used for generating hysteresis based interrupts. **Note:** The processor does not support temperature averaging.
14:14	RW	0x0	Initiate THRTMID on TEMPLO (ev_thrtmid_templo): Initiate THRTMID on TEMPLO
13:13	RW	0x1	Initiate 2X refresh on TEMPLO (ev_2x_ref_templo_en): Initiate 2X refresh on TEMPLO DIMM with extended temperature range capability will need double refresh rate in order to avoid data lost when DIMM temperature is above 85C but below 95C. **Warning:** If the 2x refresh is disable with extended temperature range DIMM configuration, system cooling and power thermal throttling scheme must guarantee the DIMM temperature will not exceed 85C.
12:12	RW	0x0	Assert MEMHOT Event on TEMPHI (ev_mh_temphi_en): Assert MEMHOT# Event on TEMPHI
11:11	RW	0x0	Assert MEMHOT Event on TEMPMID (ev_mh_tempmid_en): Assert MEMHOT# Event on TEMPMID
10:10	RW	0x0	Assert MEMHOT Event on TEMPLO (ev_mh_templo_en): Assert MEMHOT# Event on TEMPLO
9:9	RW	0x0	Assert MEMHOT Event on TEMPOEMHI (ev_mh_tempoemhi_en): Assert MEMHOT# Event on TEMPOEMHI
8:8	RW	0x0	Assert MEMHOT Event on TEMPOEMLO (ev_mh_tempoemlo_en): Assert MEMHOT# Event on TEMPOEMLO
3:0	RW	0x0	DIMM_TEMP_OFFSET (dimm_temp_offset): Bit 3-0 - Temperature Offset Register

3.3.8 dimmtempstat_[0:2]

Type: CFG Bus: 1 Offset: 0x150, 0x154, 0x158	PortID: N/A Device: 16		Function: 0,1,4,5
Bit	**Attr**	**Default**	**Description**
28:28	RW1C	0x0	Event Asserted on TEMPHI going HIGH (ev_asrt_temphi): Event Asserted on TEMPHI going HIGH It is assumed that each of the event assertion is going to trigger Configurable interrupt (Either MEMHOT# only or both SMI and MEMHOT#) defined in bit 30 of CHN_TEMP_CFG
27:27	RW1C	0x0	Event Asserted on TEMPMID going High (ev_asrt_tempmid): Event Asserted on TEMPMID going High It is assumed that each of the event assertion is going to trigger configurable interrupt (Either MEMHOT# only or both SMI and MEMHOT#) defined in bit 30 of CHN_TEMP_CFG
26:26	RW1C	0x0	Event Asserted on TEMPLO Going High (ev_asrt_templo): Event Asserted on TEMPLO Going High It is assumed that each of the event assertion is going to trigger Configurable interrupt (Either MEMHOT# only or both SMI and MEMHOT#) defined in bit 30 of CHN_TEMP_CFG
25:25	RW1C	0x0	Event Asserted on TEMPOEMLO Going Low (ev_asrt_tempoemlo): Event Asserted on TEMPOEMLO Going Low It is assumed that each of the event assertion is going to trigger Configurable interrupt (Either MEMHOT# only or both SMI and MEMHOT#) defined in bit 30 of CHN_TEMP_CFG
24:24	RW1C	0x0	Event Asserted on TEMPOEMHI Going High (ev_asrt_tempoemhi): Event Asserted on TEMPOEMHI Going High It is assumed that each of the event assertion is going to trigger Configurable interrupt (Either MEMHOT# only or both SMI and MEMHOT#) defined in bit 30 of CHN_TEMP_CFG
7:0	RW_LV	0x55	DIMM_TEMP (dimm_temp): Current DIMM Temperature for thermal throttlingLock by CLTT_DEBUG_DISABLE_LOCK When the CLTT_DEBUG_DISABLE_LOCK is cleared unlocked, debug software can write to this byte to test various temperature scenarios. When the CLTT_DEBUG_DISABLE_LOCK is set, this field becomes read-only, that is configuration write to this byte is aborted. This byte is updated from internal logic from a 2:1 multiplex which can be selected from either CLTT temperature or from the corresponding temperature registers output (PCODE_TEMP_OUTPUT) updated from pcode. The multiplex select is controlled by CLTT_OR_PCODE_TEMP_MUX_SEL defined in CHN_TEMP_CFG register. Valid range from 0 to 127 that is 0C to +127C. Any negative value read from TSOD is forced to 0. TSOD decimal point value is also truncated to integer value. The default value is changed to 85C to avoid missing refresh during S3 resume or during warm-reset flow after the DIMM is exiting self-refresh. The correct temperature may not be fetched from TSOD yet but the DIMM temperature may be still high and need to be refreshed at 2x rate.

3.3.9 thrt_pwr_dimm_[0:2]

bit[10:0]: Maximum number of transactions (ACT, READ, WRITE) to be allowed during the 1 usec throttling time frame per power throttling.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x190, 0x192, 0x194**				

Bit	Attr	Default	Description
15:15	RW	0x1	THRT_PWR_EN (thrt_pwr_en): bit[15]: set to one to enable the power throttling for the DIMM.
11:0	RW	0xfff	Power Throttling Control (thrt_pwr): bit[11:0]: Maximum number of transactions (ACT, READ, WRITE) to be allowed (per DIMM) during the 1 μ-second throttling timeframe per power throttling. PCODE can update this register dynamically.

3.3.10 tcdbp

Timing Constraints DDR3 Bin Parameter.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x200**				

Bit	Attr	Default	Description
26:26	RW	0x0	cmd_oe_cs:
25:25	RW	0x0	cmd_oe_on:
24:19	RW	0x1c	T_RAS (t_ras):
18:14	RW	0x7	T_CWL (t_cwl):
13:9	RW	0xa	T_CL (t_cl):
8:5	RW	0xa	T_RP (t_rp):
4:0	RW	0xa	T_RCD (t_rcd):

3.3.11 tcrap

Timing Constraints DDR3 Regular Access Parameter.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x204**				

Bit	Attr	Default	Description
31:30	RW	0x0	CMD_STRETCH (cmd_stretch): defines for how many cycles the command is stretched 00: 1N operation 01: Reserved 10: 2N operation 11: 3N operation
29:29	RW	0x0	CMD_3ST (cmd_3st): When cleared, command and address is driving only when required. When set, command and address are driving always, and the value when no valid command is the last command and address
28:24	RW	0xc	T_WR (t_wr): WRITE recovery time (must be at least 15ns equivalent)
23:22	RW	0x1	T_PRPDEN (t_prpden): tPRPDEN, tACTPDEN, tREFPDEN needs to have value of 2 for 2133. All 3 values will use this single value.
21:16	RW	0x20	T_FAW (t_faw): Four activate window (must be at least 4 * tRRD and at most 63)
15:12	RW	0x6	T_WTR (t_wtr): DCLK delay from start of internal write transaction to internal read command (must be at least the larger value of 4 DCLK or 7.5ns) iMC's Write to Read Same Rank (T_WRSR) is automatically calculated based from TCDBP.T_CWL + 4 + T_WTR. For LRDIMM running in rank multiplication mode, iMC will continue to use the above equation for T_WRSR even if the WRITE and READ are targeting same logical rank but at different physical ranks behind the LRDIMM buffer, In the other word, iMC will not be able to dynamically switch to TWRDR timing. In order to avoid timing violation in this scenario, BIOS must configure the TWTR parameter to be the MAX (T_WTR of LRDIMM, (T_WRDR' - TCL + 2)). **Note:**: Due to the lighter electrical loading behind the LRDIMM buffer, further optimization can be tuned during post-silicon to reduce the T_WRDR' parameter instead of directly using the TCRWP.T_WRDR parameter.
11:8	RW	0x3	T_CKE (t_cke): CKE minimum pulse width (must be at least the larger value of 3 DCLK or 5ns)
7:4	RW	0xa	T_RTP (t_rtp): Internal READ Command to PRECHARGE Command delay, (must be at least the larger value of 4 DCLK or 7.5ns)
2:0	RW	0x5	T_RRD (t_rrd): ACTIVE to ACTIVE command period, (must be at least the larger value of 4 DCLK or 6ns)

3.3.12 tcrwp

Timing Constraints DDR3 Read Write Parameter.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x208**				

Bit	Attr	Default	Description
31:30	RW	0x0	T_WRDR_UPPER (t_wrdr_upper): Upper 2 bits, bits 4:3 of twrdr field.
29:27	RW	0x0	T_CCD (t_ccd): back to back READ to READ or CAS to CAS from same rank separation parameter.The actual JEDEC CAS to CAS command separation is (T_CCD + 4) DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#.
26:24	RW	0x2	T_RWSR (t_rwsr): This field is not used in the processor. Refer to TCOTHP2 for the new register field location.
23:21	RW	0x2	T_WRDD (t_wrdd): Back to back WRITE to READ from different DIMM separation parameter.The actual WRITE to READ command separation is TCDBP.T_CWL - TCDBP.TCL + T_WRDD + 6 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#.
20:18	RW	0x2	T_WRDR (t_wrdr): Back to back WRITE to READ from different RANK separation parameter.The actual WRITE to READ command separation is TCDBP.T_CWL - TCDBP.TCL + T_WRDR + 6 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#.
17:15	RW	0x2	T_RWDD (t_rwdd): This field is not used starting in the processor. Refer to TCOTHP2 for the new register field location.
14:12	RW	0x2	T_RWDR (t_rwdr): This field is not used starting in the processor. Refer to TCOTHP2 for the new register field location.
11:9	RW	0x2	T_WWDD (t_wwdd): Back to back WRITE to WRITE from different DIMM separation parameter. The actual WRITE to WRITE command separation is T_WWDD + 5 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#. **Note:** The minimum setting of the field must meet the DDRIO requirement for WRITE to WRITE turnaround time to be at least 6 DClk at the DDRIO pin. The maximum design range from the above calculation is 15.

Type: CFG Bus: 1 Offset: 0x208		PortID: N/A Device: 16	Function: 0,1,4,5
Bit	**Attr**	**Default**	**Description**
8:6	RW	0x2	T_WWDR (t_wwdr): Back to back WRITE to WRITE from different RANK separation parameter. The actual WRITE to WRITE command separation is T_WWDR + 5 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#. **Note:** The minimum setting of the field must meet the DDRIO requirement for WRITE to WRITE turnaround time to be at least 6 DClk at the DDRIO pin. The maximum design range from the above calculation is 15.
5:3	RW	0x2	T_RRDD (t_rrdd): Back to back READ to READ from different DIMM separation parameter. The actual READ to READ command separation is TRRDD + 5 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#. **Note:** The minimum setting of the field must meet the DDRIO requirement for READ to READ turnaround time to be at least 5 DClk at the DDRIO pin. The maximum design range from the above calculation is 31.
2:0	RW	0x2	T_RRDR (t_rrdr): Back to back READ to READ from different RANK separation parameter. The actual READ to READ command separation is TRRDR + 5 DCLKs measured between the clock assertion edges of the two corresponding asserted command CS#. **Note:** The minimum setting of the field must meet the DDRIO requirement for READ to READ turnaround time to be at least 5 DClk at the DDRIO pin. The maximum design range from the above calculation is 31.

3.3.13 tcothp

Timing Constraints DDR3 Other Timing Parameter.

Type: CFG Bus: 1 Offset: 0x20c		PortID: N/A Device: 16	Function: 0,1,4,5
Bit	**Attr**	**Default**	**Description**
31:28	RW	0x6	t_cs_oe: When tcsoe0, CS9:0# will not tri-state Otherwise, this field defines delay in Dclks to disable CS output after all CKE pins are low
27:24	RW	0x6	t_odt_oe: When todtoe0, ODT will not tri-state Otherwise, this field defines delay in Dclks to disable ODT output after all CKE pins are low and either in self-refresh or in IBTOff mode
23:20	RW	0x0	t_rwsr: Do not use for this processor
19:16	RW	0x0	t_rwdd: Do not use for this processor
15:12	RW	0x2	t_rwdr:
11:11	RW	0x0	shift_odt_early: This shifts the ODT waveform one cycle early relative to the timing set up in the ODT_TBL2 register, when in 2N or 3N mode. This bit has no effect in 1N mode.

Type: CFG Bus: 1 Offset: 0x20c		PortID: N/A Device: 16	Function: 0,1,4,5
Bit	**Attr**	**Default**	**Description**
10:8	RW	0x0	T_CWL_ADJ (t_cwl_adj): This register defines additional WR data delay per channel in order to overcome the WR-flyby issue. The total CAS write latency that the DDR sees is the sum of T_CWL and the T_CWL_ADJ. 000 - no added latency default 001 - 1 Dclk of added latency 010 - 2 Dclk of added latency 011 - 3 Dclk of added latency 1xx - Reduced latency by 1 Dclk. Not supported at tCWL = 5
7:5	RW	0x3	T_XP (t_xp): Exit Power Down with DLL on to any valid command; Exit Precharge Power Down with DLL frozen to commands not requiring a locked DLL.
4:0	RW	0xa	T_XPDLL (t_xpdll): Exit Precharge Power Down with DLL frozen to commands requiring a locked DLL.

3.3.14 tcrfp

Timing Constraints DDR3 Refresh Parameter.

Type: CFG Bus: 1 Offset: 0x210		PortID: N/A Device: 16	Function: 0,1,4,5
Bit	**Attr**	**Default**	**Description**
15:12	RW	0x9	REF_PANIC_WM (ref_panic_wm): tREFI count level in which the refresh priority is panic (default is 9) It is recommended to set the panic WM at least to 9, in order to utilize the maximum no-refresh period possible
11:8	RW	0x8	REF_HI_WM (ref_hi_wm): tREFI count level that turns the refresh priority to high (default is 8)
7:0	RW	0x3f	OREFNI (orefni): Rank idle period that defines an opportunity for refresh, in DCLK cycles

3.3.15 tcrftp

Timing Constraints Refresh Timing Parameter.

Type: CFG **Bus:** 1 **Offset:** 0x214		**PortID:** N/A **Device:** 16	**Function:** 0,1,4,5
Bit	**Attr**	**Default**	**Description**
31:25	RW	0x9	T_REFIX9 (t_refix9): period of minimum between 9 * T_REFI and tRAS maximum (normally 70 µ-sec) in 1024 * DCLK cycles.The default value will need to reduce 100 DCLK cycles - uncertainty on timing of panic refresh
24:15	RW	0x80	T_RFC (t_rfc): Time of refresh - from beginning of refresh until next ACT or refresh is allowed (in DCLK cycles) Here are the recommended T_RFC for 2Gb DDR3: 0800 MT/s : 040h 1067 MT/s : 056h 1333 MT/s : 06Bh 1600 MT/s : 080h 1867 MT/s : 096h
14:0	RW	0x62c	T_REFI (t_refi): Defines the average period between refreshes in DCLK cycles. This register defines the upper 15b of the 16b tREFI counter limit. The least significant bit of the counter limit is always zero. Here are the recommended T_REFI[14:0] setting for 7.8 µ-sec: 0800 MT/s : 0C30h 1067 MT/s : 1040h 1333 MT/s : 1450h 1600 MT/s : 1860h 1867 MT/s : 1C70h

3.3.16 tcsrftp

Timing Constraints Self-Refresh Timing Parameter.

Type: CFG **Bus:** 1 **Offset:** 0x218		**PortID:** N/A **Device:** 16	**Function:** 0,1,4,5
Bit	**Attr**	**Default**	**Description**
31:27	RW	0xc	T_MOD (t_mod): Mode Register Set command update delay.
25:16	RW	0x100	T_ZQOPER (t_zqoper): Normal operation Full calibration time

15:12	RW	0xb	T_XSOFFSET (t_xsoffset):
			tXS = T_RFC + 10ns. Setup of T_XSOFFSET is # of cycles for 10 ns. Range is between 3 and 11 DCLK cycles

11:0	RW	0x100	T_XSDLL (t_xsdll): Exit Self Refresh to commands requiring a locked DLL in the range of 128 to 4095 DCLK cycles

3.3.17 tcmr2shadow

Timing Constraints MR2 Shadow Timing Parameter

Type: **CFG** **Bus:** **1** **Offset:** **0x21c**	**PortID:** **N/A** **Device:** **16**	**Function:** **0,1,4,5**	
Bit	**Attr**	**Default**	**Description**
26:24	RW_LV	0x0	ADDR_BIT_SWIZZLE (addr_bit_swizzle): Each bit is set in case of the corresponding 2-rank UDIMM requires address swizzling. It indicates that some of the address bits are swizzled for rank 1 (or rank 3), and this has to be considered in MRS command. The address swizzling bits: A3 and A4 A5 and A6 A7 and A8 BA0 and BA1 Bit 24 refers to DIMM 0 Bit 25 refers to DIMM 1 Bit 26 refers to DIMM 2
23:16	RW	0x2	MR2_SHDW_A15TO8 (mr2_shdw_a15to8): Copy of MR2 A[15:8] shadow. Bit 23-19: zero, copy of MR2 A[15:11], reserved for future JEDEC use Bit 18-17: Rtt_WR, that is, copy of MR2 A[10:9] Bit 16: zero, copy of MR2 A[8], reserved for future JEDEC use
14:12	RW	0x0	MR2_SHDW_A7_SRT (mr2_shdw_a7_srt): Copy of MR2 A[7] shadow which defines per DIMM availability of SRT mode - set if extended temperature range and ASR is not supported, otherwise cleared Bit 14: Dimm 2 Bit 13: Dimm 1 Bit 12: Dimm 0
10:8	RW	0x0	MR2_SHDW_A6_ASR (mr2_shdw_a6_asr): Copy of MR2 A[6] shadow which defines per DIMM availability of ASR mode - set if Auto Self-Refresh (ASR) is supported, otherwise cleared Bit 10: Dimm 2 Bit 9: Dimm 1 Bit 8: Dimm 0
5:0	RW	0x18	MR2_SHDW_A5TO0 (mr2_shdw_a5to0): Copy of MR2 A[5:0] shadow

3.3.18 tczqcal

Timing Constraints ZQ Calibration Timing Parameter

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x220**				

Bit	Attr	Default	Description
15:8	RW	0x40	T_ZQCS (t_zqcs): tZQCS in DCLK cycles (32 to 255, default is 64)
7:0	RW	0x80	ZQCSPERIOD (zqcsperiod): Time between ZQ-FSM initiated ZQCS operations in tREFI * 128 (2 to 255, default is 128). **Note**: ZQCx is issued at SRX

3.3.19 tcstagger_ref

tRFC like timing constraint parameter except it is a timing constraint applicable to REF-REF separation between different ranks on a channel.

Note: This register value only become effective after MCMNT_UCR_CHKN_BIT.STAGGER_REF_EN is set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x224**				

Bit	Attr	Default	Description
9:0	RW	0x80	T_STAGGER_REF (t_stagger_ref): tRFC like timing constraint parameter except it is a timing constraint applicable to REF-REF separation between different ranks on a channel. It is recommended to set T_STAGGER_REF equal or less than the T_RFC parameter which is defined as: 0800MT/s : 040h 1067MT/s : 056h 1333MT/s : 06Bh 1600MT/s : 080h 1867MT/s : 096h

3.3.20 tcmr0shadow

MR0 Shadow Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x22c**				

Bit	Attr	Default	Description
11:0	RW	0x0	MR0_SHADOW (mr0_shadow): BIOS program this field for MR0 register A11:A0 for all DIMMs in this channel. iMC hardware is dynamically issuing MRS to MR0 to control the fast and slow exit PPD (MRS MR0 A12). Other address bits (A[11:0]) is defined by this register field. A15:A13 are always zero.

3.3.21 rpqage

Read Pending Queue Age Counters.

Type: CFG **Bus:** 1 **Offset:** 0x234	**PortID:** N/A **Device:** 16		**Function:** 0,1,4,5
Bit	**Attr**	**Default**	**Description**
25:16	RW	0x0	IOCount (iocount): This is the RPQ Age Counter for the Medium and Low priority (VC0) non-isoch transactions issued from HA. The counter is increased by one every time there's a CAS command sent. When the RPQ Age Counter is equal to this configured field value, the non-isoch transaction is aged to the next priority level. BIOS must set this field to non-zero value before setting the MCMTR.NORMAL = 1.
9:0	RW	0x0	Reserved

3.3.22 idletime

At a high level, the goal of any page closing policy is to trade off some Premature Page Closes (PPCs) in order to avoid more Overdue Page Closes (OPCs). In other words, we want to avoid costly Page Misses and turn them into Page Empties at the expense of occasionally missing a Page Hit and instead getting a Page Empty. The scheme achieves this by tracking the number of PPCs and OPCs over a certain configurable window (of requests). It then compares the two values to configurable thresholds, and adjusts the amount of time before closing pages accordingly.

Type: CFG **Bus:** 1 **Offset:** 0x238	**PortID:** N/A **Device:** 16		**Function:** 0,1,4,5
Bit	**Attr**	**Default**	**Description**
28:28	RW	0x1	ADAPT_PG_CLSE (adapt_pg_clse): This register is programmed in conjunction with MCMTR.CLOSEPG to enable three different modes: 1 Closed Page Mode Mode -- MCMTR.CLOSE_PG = 1 and ADAPT_PG_CLSE = 0 2 Open Page Mode Mode -- MCMTR.CLOSE_PG = 0 and ADAPT_PG_CLSE = 0 3 Adaptive Open Open -- MCMTR.CLOSE_PG = 0 and ADAPT_PG_CLSE = 1 MCMTR.CLOSE_PG = 1 and ADAPT_PG_CLSE = 1 is illegal. When ADAPT_PG_CLSE = 0, the page close idle timer gets set with IDLE_PAGE_RST_VAL times 4.
27:21	RW	0x6	OPC_TH (opc_th): Overdue Page Close (OPC) Threshold If the number of OPCs in a given window is larger than this threshold, we decrease the RV.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x238**				

Bit	Attr	Default	Description
20:14	RW	0x6	PPC_TH (ppc_th): Premature Page Close (PPC) Threshold If the number of PPCs in a given window is larger than this threshold, we increase the RV
13:6	RW	0x40	WIN_SIZE (win_size): Window Size (WS): The number of requests that we track before making a decision to adapt the RV.
5:0	RW	0x8	IDLE_PAGE_RST_VAL (idle_page_rst_val): Idle Counter Reset Value (RV): This is the value that effectively adapts. It determines what value the various ICs are set to whenever they are reset. It therefore controls the number of cycles before an automatic page close is triggered for an entire channel.

3.3.23 rdimmtimingcntl

RDIMM Timing Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**0,1,4,5**
Offset:	**0x23c**				

Bit	Attr	Default	Description
28:16	RW	0x12c0	T_STAB (t_stab): Stabilizing time in number of DCLK, that is, the DCLK must be stable for T_STAB before any access to the device take place. We have included tCKSRX in the T_STAB programming since processor does not have separate tCKSRX parameter control to delay self-refresh exit latency from clock stopped conditions. **Note** #1: zero value in T_STAB is reserved and it is important to AVOID programming a zero value in the T_STAB. Recommended settings **Note**: Contains stretch goal and/or over-clock frequency examples: FREQ TSTAB for RDIMM (including tCKSRX value) 0800 0960h + 5h = 0965h 1067 0C80h + 5h = 0c85h 1333 0FA0h + 7h = 0FA7h 1600 12C0h + 8h = 12C8h 1867 15E0h + Ah = 15EAh 2133 1900h + Bh = 190Bh FREQ TSTAB for UDIMM (that is, tCKSRX value) 0800 7h 1067 7h 1333 9h 1600 Ah 1867 Ch 2133 Dh
3:0	RW	0x8	T_MRD (t_mrd): Command word to command word programming delay in DCLK

3.3.24 rdimmtimingcntl2

Type:	CFG	PortID:	N/A		
Bus:	1	Device:	16	Function:	0,1,4,5
Offset:	0x240				

Bit	Attr	Default	Description
7:4	RW	0x2	T_CKEV (t_ckev): Input buffers DCKE0 and DCKE1 disable time float after CK/CK# LOW not needed since DDRIO cannot program the CKE to be tri-stated.
3:0	RW	0x5	T_CKOFF (t_ckoff): tCKOFF timing parameter: Number of tCK required for both DCKE0 and DCKE1 to remain LOW before both CK/CK# are driven Low Minimum setting is 2.

3.3.25 tcmrs

Type:	CFG	PortID:	N/A		
Bus:	1	Device:	16	Function:	0,1,4,5
Offset:	0x244				

Bit	Attr	Default	Description
3:0	RW	0x8	TMRD_DDR3 (tmrd_ddr3): DDR3 tMRD timing parameter. MRS to MRS minimum delay in number of DCLK.

3.3.26 mc_init_stat_c

State register per channel. Sets control signals static values. Power-up default is state 0x0 set by global reset.

BIOS should leave this register default to zero since PCODE has ReadWrite ODT table logic to control ODT dynamically during IOSAV or NORMAL modes.

Type:	CFG	PortID:	N/A		
Bus:	1	Device:	16	Function:	0,1,4,5
Offset:	0x280				

Bit	Attr	Default	Description
13:8	RW-L	0x0	ODT override (odt_override): When set, the bit overrides and asserts the corresponding ODT[5:0] output signal during IOSAV mode. When cleared, the ODT pin is controlled by the IMC IOSAV logic.
5:0	RW-L	0x0	CKE ON OVERRIDE (cke_on): When set, the bit overrides and asserts the corresponding CKE[5:0] output signal during IOSAV mode. When cleared, the CKE pin is controled by the IMC IOSAV logic.

3.4 Device 16 Functions 2, 3, 7

Table 3-4. IMC Device 16 Functions 2, 3, 7 Register Address Map

Register Name	Offset	Size	Functions
correrrcnt_0	0x104	32	2,3,7
correrrcnt_1	0x108	32	2,3,7
correrrcnt_2	0x10c	32	2,3,7
correrrcnt_3	0x110	32	2,3,7
correrrthrshld_0	0x11c	32	2,3,7
correrrthrshld_1	0x120	32	2,3,7
correrrthrshld_2	0x124	32	2,3,7
correrrthrshld_3	0x128	32	2,3,7
correrrorstatus	0x134	32	2,3,7
leaky_bkt_2nd_cntr_reg	0x138	32	2,3,7
devtag_cntl_0	0x140	8	2,3,7
devtag_cntl_1	0x141	8	2,3,7
devtag_cntl_2	0x142	8	2,3,7
devtag_cntl_3	0x143	8	2,3,7
devtag_cntl_4	0x144	8	2,3,7
devtag_cntl_5	0x145	8	2,3,7
devtag_cntl_6	0x146	8	2,3,7
devtag_cntl_7	0x147	8	2,3,7

3.4.1 correrrcnt_0

Per Rank corrected error counters.

Type: CFG **Bus:** 1 **Offset:** 0x104	**PortID:** N/A **Device:** 16	**Function:** 2,3,7	
Bit	**Attr**	**Default**	**Description**
31:31	RW1CS	0x0	RANK 1 OVERFLOW (overflow_1): The corrected error count for this rank has been overflowed. Once set it can only be cleared by means of a write from BIOS.
30:16	RWS_V	0x0	RANK 1 CORRECTABLE ERROR COUNT (cor_err_cnt_1): The corrected error count for this rank. Hardware automatically clears this field when the corresponding OVERFLOW_x bit is changing from 0 to 1.
15:15	RW1CS	0x0	RANK 0 OVERFLOW (overflow_0): The corrected error count for this rank has been overflowed. Once set it can only be cleared by means of a write from BIOS.
14:0	RWS_V	0x0	RANK 0 CORRECTABLE ERROR COUNT (cor_err_cnt_0): The corrected error count for this rank. Hardware automatically clear this field when the corresponding OVERFLOW_x bit is changing from 0 to 1.

3.4.2 correrrcnt_1

Per Rank corrected error counters.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x108**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	RANK 3 OVERFLOW (overflow_3): The corrected error count has crested over the limit for this rank. Once set it can only be cleared by means of a write from BIOS.
30:16	RWS_V	0x0	RANK 3 COR_ERR_CNT (cor_err_cnt_3): The corrected error count for this rank.
15:15	RW1CS	0x0	RANK 2 OVERFLOW (overflow_2): The corrected error count has crested over the limit for this rank. Once set it can only be cleared by means of a write from BIOS.
14:0	RWS_V	0x0	RANK 2 COR_ERR_CNT (cor_err_cnt_2): The corrected error count for this rank.

3.4.3 correrrcnt_2

Per Rank corrected error counters.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x10c**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	RANK 5 OVERFLOW (overflow_5): The corrected error count has crested over the limit for this rank. Once set it can only be cleared by means of a write from BIOS.
30:16	RWS_V	0x0	RANK 5 COR_ERR_CNT (cor_err_cnt_5): The corrected error count for this rank.
15:15	RW1CS	0x0	RANK 4 OVERFLOW (overflow_4): The corrected error count has crested over the limit for this rank. Once set it can only be cleared by means of a write from BIOS.
14:0	RWS_V	0x0	RANK 4 COR_ERR_CNT (cor_err_cnt_4): The corrected error count for this rank.

3.4.4 correrrcnt_3

Per Rank corrected error counters.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x110**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	RANK 7 OVERFLOW (overflow_7): The corrected error count for this rank.
30:16	RWS_V	0x0	RANK 7 COR_ERR_CNT_7 (cor_err_cnt_7): The corrected error count for this rank.
15:15	RW1CS	0x0	RANK 6 OVERFLOW (overflow_6): The corrected error count has crested over the limit for this rank. Once set it can only be cleared by means of a write from BIOS.
14:0	RWS_V	0x0	RANK 6 COR_ERR_CNT (cor_err_cnt_6): The corrected error count for this rank.

3.4.5 correrrthrshld_0

This register holds the per rank corrected error thresholding value.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x11c**				

Bit	Attr	Default	Description
30:16	RW	0x7fff	RANK 1 COR_ERR_TH (cor_err_th_1): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.
14:0	RW	0x7fff	RANK 0 COR_ERR_TH (cor_err_th_0): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.

3.4.6 correrrthrshld_1

This register holds the per rank corrected error thresholding value.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x120**				

Bit	Attr	Default	Description
30:16	RW	0x7fff	RANK 3 COR_ERR_TH (cor_err_th_3): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.
14:0	RW	0x7fff	RANK 2 COR_ERR_TH (cor_err_th_2): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.

3.4.7 correrrthrshld_2

This register holds the per rank corrected error thresholding value.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x124**				

Bit	Attr	Default	Description
30:16	RW	0x7fff	RANK 5 COR_ERR_TH (cor_err_th_5): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.
14:0	RW	0x7fff	RANK 4 COR_ERR_TH (cor_err_th_4): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.

3.4.8 correrrthrshld_3

This register holds the per rank corrected error thresholding value.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x128**				

Bit	Attr	Default	Description
30:16	RW	0x7fff	RANK 7 COR_ERR_TH (cor_err_th_7): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.
14:0	RW	0x7fff	RANK 6 COR_ERR_TH (cor_err_th_6): The corrected error threshold for this rank that will be compared to the per rank corrected error counter.

3.4.9 correrrorstatus

Per rank corrected error status. These bits are reset by BIOS.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x134**				

Bit	Attr	Default	Description
7:0	RW1C	0x0	ERR_OVERFLOW_STAT (err_overflow_stat): This 8-bit field is the per rank error over-threshold status bits. The organization is as follows: Bit 0 : Rank 0 Bit 1 : Rank 1 Bit 2 : Rank 2 Bit 3 : Rank 3 Bit 4 : Rank 4 Bit 5 : Rank 5 Bit 6 : Rank 6 Bit 7 : Rank 7 **Note**: The register tracks which rank has reached or exceeded the corresponding CORRERRTHRSHLD threshold settings.

3.4.10 leaky_bkt_2nd_cntr_reg

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**16**	**Function:**	**2,3,7**
Offset:	**0x138**				

Bit	Attr	Default	Description
31:16	RW	0x0	LEAKY_BKT_2ND_CNTR_LIMIT (leaky_bkt_2nd_cntr_limit): Secondary Leaky Bucket Counter Limit (2b per DIMM). This register defines secondary leaky bucket counter limit for all 8 logical ranks within channel. The counter logic will generate the secondary LEAK pulse to decrement the rank's correctable error counter by 1 when the corresponding rank leaky bucket rank counter roll over at the predefined counter limit. The counter increment at the primary leak pulse from the LEAKY_BUCKET_CNTR_LO and LEAKY_BUCKET_CNTR_HI logic. Bit[31:30]: Rank 7 Secondary Leaky Bucket Counter Limit Bit[29:28]: Rank 6 Secondary Leaky Bucket Counter Limit Bit[27:26]: Rank 5 Secondary Leaky Bucket Counter Limit Bit[25:24]: Rank 4 Secondary Leaky Bucket Counter Limit Bit[23:22]: Rank 3 Secondary Leaky Bucket Counter Limit Bit[21:20]: Rank 2 Secondary Leaky Bucket Counter Limit Bit[19:18]: Rank 1 Secondary Leaky Bucket Counter Limit Bit[17:16]: Rank 0 Secondary Leaky Bucket Counter Limit The value of the limit is defined as the following: 0: the LEAK pulse is generated one DCLK after the counter roll over at 3. 1: the LEAK pulse is generated one DCLK after the primary LEAK pulse is asserted. 2: the LEAK pulse is generated one DCLK after the counter roll over at 1. 3: the LEAK pulse is generated one DCLK after the counter roll over at 2.
15:0	RW-V	0x0	LEAKY_BKT_2ND_CNTR (leaky_bkt_2nd_cntr): Per rank secondary leaky bucket counter (2b per rank) bit [15:14]: rank 7 secondary leaky bucket counter bit [13:12]: rank 6 secondary leaky bucket counter bit [11:10]: rank 5 secondary leaky bucket counter bit [9:8]: rank 4 secondary leaky bucket counter bit [7:6]: rank 3 secondary leaky bucket counter bit [5:4]: rank 2 secondary leaky bucket counter bit [3:2]: rank 1 secondary leaky bucket counter bit [1:0]: rank 0 secondary leaky bucket counter

3.4.11 devtag_cntl_[0:7]

SDDC Usage model

When the number of correctable errors (CORRERRCNT_x) from a particular rank exceeds the corresponding threshold (CORRERRTHRSHLD_y), hardware will generate a SMI interrupt and log and preserve the failing device in the FailDevice field. SMM software will read the failing device on the particular rank. Software then set the EN bit to enable substitution of the failing device/rank with the parity from the rest of the devices in line.

For independent channel configuration, each rank can tag once. Up to 8 ranks can be tagged.

For lock-step channel configuration, only one x8 device can be tagged per rank-pair. SMM software must identify which channel should be tagged for this rank and only set the valid bit for the channel from the channel-pair.

There is no hardware logic to report incorrect programming error. Unpredicable error and or silent data corruption will be the consequence of such programming error.

Type: CFG **Bus:** 1 **Offset:** 0x140, 0x141, 0x142, 0x143, 0x144, 0x145, 0x146, 0x147		**PortID:** N/A **Device:** 16	**Function:** 2,3,7
Bit	**Attr**	**Default**	**Description**
7:7	RWS_L	0x0	Device tagging enable for this rank (en): Device tagging SDDC enable for this rank. Once set, the parity device of the rank is used for the replacement device content. After tagging, the rank will no longer have the “correction” capability. Warning: For lock-step channel configuration, only one x8 device can be tagged per rank-pair. SMM software must identify which channel should be tagged for this rank and only set the corresponding DEVTAG_CNTL_x.EN bit for the channel contains the fail device. The DEVTAG_CNTL_x.EN on the other channel of the corresponding rank must not be set. Must never be enable prior using IOSAV DDDC: On DDDC supported systems, BIOS has the option to enable SDDC in conjunction with DDDC_CNTL:SPARING to enable faster sparing with SDDC substitution. This field is cleared by hardware on completion of DDDC sparing.
5:0	RWS_V	0x3f	Fail Device ID for this rank (faildevice): Hardware will capture the fail device ID of the rank in the FailDevice field upon successful correction from the device correction engine. After SDDC is enabled hardware may not update this field. Valid Range is decimal 0-17 to indicate which x4 device (independent channel) or x8 device (lock-step mode) has failed. Valid Range is decimal 0-35 to indicate which x4 device has failed. **Note**: When DDDC has been enabled on the non-spare device, and a subsequent failure of the spare device occurs, the value logged here will be equal to the DDDC faildevice.

§ §

4 R2PCIe

4.1 Device 19 Function 0

DID		VID		0h		80h
PCISTS		PCICMD		4h		84h
CCR			RID	8h		88h
BIST	HDR	PLAT	CLSR	Ch		8Ch
				10h		90h
				14h		94h
				18h		98h
				1Ch		9Ch
				20h		A0h
				24h		A4h
				28h		A8h
SDID		SVID		2Ch		ACh
				30h		B0h
			CAPPTR	34h		B4h
				38h		B8h
MAXLAT	MINGNT	INTPIN	INTL	3Ch		BCh
				40h		C0h
				44h		C4h
				48h		C8h
				4Ch		CCh
				50h		D0h
				54h		D4h
				58h		D8h
				5Ch		DCh
				60h		E0h
				64h		E4h
				68h		E8h
				6Ch		ECh
				70h		F0h
				74h		F4h
				78h		F8h
				7Ch		FCh

§ §

5 Processor Utility Box (UBOX) Registers

The UBOX is the piece of processor logic that deals with the non mainstream flows in the system. This includes transactions like the register accesses, interrupt flows, lock flows and events. In addition, the UBOX houses coordination for the performance architecture, and also houses scratchpad and semaphore registers.

5.1 Device 11 Function 0

Table 5-1. UBOX Device 11 Function 0 Register Address Map

Register Name	Offset	Size
CPUNODEID	0x40	32
IntControl	0x48	32
GIDNIDMAP	0x54	32
CoreCount	0x60	32
UBOXErrSts	0x64	32
BIOSStickyLockBypassScratchpad1	0x78	32
BIOSStickyLockBypassScratchpad2	0x7c	32
BIOSStickyLockBypassScratchpad3	0x80	32
BIOSStickyLockBypassScratchpad4	0x84	32
BIOSStickyLockBypassScratchpad5	0x88	32
BIOSStickyLockBypassScratchpad6	0x8c	32
UBOX_GL_ERR_CFG	0x90	32

5.1.1 CPUNODEID

Node ID Configuration Register

Type: CFG **Bus:** 1 **Offset:** 0x40			**PortID:** N/A **Device:** 11 **Function:** 0
Bit	**Attr**	**Default**	**Description**
15:13	RW-LB	0x0	Node Controller Node Id(NodeCtrlId): Node ID of the Node Controller. Set by the BIOS.
12:10	RW-LB	0x0	NodeID of the legacy socket(LgcNodeId): NodeID of the legacy socket
7:5	RW-LB	0x0	NodeId of the lock master(LockNodeId): NodeId of the lock master
2:0	RW-LB	0x0	NodeId of the local register(LclNodeId): Node Id of the local socket

5.1.2 IntControl

Interrupt Configuration Register

Type: CFG **Bus:** 1 **Offset:** 0x48			**PortID:** N/A **Device:** 11 **Function:** 0
Bit	**Attr**	**Default**	**Description**
18:18	RW-LB	0x0	IA-32 Logical Flat or Cluster Mode Override Enable(LogFlatClustOvrEn): 0 : IA-32 Logical Flat or Cluster Mode bit is locked as Read only bit. 1 : IA-32 Logical Flat or Cluster Mode bit may be written by software, values written by xTPR update are ignored. For one time override of the IA-32 Logical Flat or Cluster Mode value, return this bit to it's default state after the bit is changed. Leaving this bit as '1' will prevent automatic update of the filter.
17:17	RW_LBV	0x0	IA-32 Logical Flat or Cluster Mode(LogFltClustMod): Set by BIOS to indicate if the OS is running logical flat or logical cluster mode. This bit can also be updated by IntPrioUpd messages. This bit reflects the setup of the filter at any given time. 0 - flat, 1 - cluster.
16:16	RW-LB	0x0	Cluster Check Sampling Mode(ClastChkSmpMod): 0: Disable checking for Logical_APICID[31:0] being non-zero when sampling flat cluster mode bit in the IntPrioUpd message as part of setting bit 1 in this register 1: Enable the above checking
10:8	RW-LB	0x0	Vecor Based Hashe Mode Control(HashModCtr): Indicates the hash mode control for the interrupt control. Select the hush function for the Vector based Hash Mode interrupt redirection control : 000 select bits 7:4/5:4 for vector cluster/flat algorithm 001 select bits 6:3/4:3 010 select bits 4:1/2:1 011 select bits 3:0/1:0 other - reserved

Type: CFG Bus: 1 Offset: 0x48		PortID: N/A Device: 11	Function: 0
Bit	**Attr**	**Default**	**Description**
6:4	RW-LB	0x0	Redirection Mode Select for Logical Interrupts(RdrModSel): Selects the redirection mode used for MSI interrupts with lowest-priority delivery mode. The following schemes are used: 000 : Fixed Priority - select the first enabled APIC in the cluster. 001: Redirect last - last vector selected (applicable only in extended mode) 010 : Hash Vector - select the first enabled APIC in round robin manner starting form the hash of the vector number. 100: Fixed Priority - with PLAIR. 101: Redirect Last - with PLAIR. 110: Hash Vector - with PLAIR. default: Fixed Priority
1:1	RW-LB	0x0	Force to X2 APIC Mode(ForceX2APIC): Write: 1: Forces the system to move into X2APIC Mode. 0: No affect
0:0	RW-LB	0x1	Extended APIC Enable(xApicEn): Set this bit if you would like extended XAPIC configuration to be used. This bit can be written directly, and can also be updated using XTPR messages

5.1.3 GIDNIDMAP

Node ID Mapping Register.

Mapping between group id and nodeid

Type: CFG Bus: 1 Offset: 0x54		PortID: N/A Device: 11	Function: 0
Bit	**Attr**	**Default**	**Description**
23:21	RW-LB	0x0	Node Id 7(NodeId7): NodeId for group id 7
20:18	RW-LB	0x0	Node Id 6(NodeId6): Node Id for group 6
17:15	RW-LB	0x0	Node Id 5(NodeId5): Node Id for group 5
14:12	RW-LB	0x0	Node Id 4(NodeId4): Node Id for group id 4
11:9	RW-LB	0x0	Node Id 3(NodeId3): Node Id for group 3
8:6	RW-LB	0x0	Node Id 2(NodeID2): Node Id for group Id 2
5:3	RW-LB	0x0	Node Id 1(NodeId1): Node Id for group Id 1
2:0	RW-LB	0x0	Node Id 0(NodeId0): Node Id for group 0

5.1.4 CoreCount

Number of Cores

Reflection of the LTCount2 register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**11**	**Function:**	**0**
Offset:	**0x60**				

Bit	Attr	Default	Description
4:0	RO-V	0x0	Core Count(CoreCount): Reflection of the LTCount2

5.1.5 UBOXErrSts

This is error status register in the UBOX and covers most of the interrupt related errors.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**11**	**Function:**	**0**
Offset:	**0x64**				

Bit	Attr	Default	Description
23:18	RWS_V	0x0	Message Channel Tracker TimeOut(Msg_Ch_Tkr_TimeOut): Message Channel Tracker TimeOut. This error occurs when any NP request doesn't receive response in 4K cycles. The event is SV use and logging only, not signaling
17:17	RWS_V	0x0	Message Channel Tracker Error(Msg_Ch_Tkr_Err): Message Channel Tracker Error. This error occurs such case that illegal broad cast port ID access to the message channel. The event is SV use and logging only, not signaling as Ubox error.
16:16	RW-V	0x0	SMI delivery valid(SMI_delivery_valid): SMI interrupt delivery status valid, write 1'b1 to clear valid status
15:8	RO-V	0x0	reserved:
7:7	RWS_V	0x0	MasterLock Timeout received by UBOX(MasterLockTimeOut): Master Lock Timeout received by UBOX
6:6	RWS_V	0x0	SMI Timeout received by UBOX(SMITimeOut): SMI Timeout received by UBOX
5:5	RWS_V	0x0	MMCFG Write Address Misalignment received by UBOX(CFGWrAddrMisAligned): MMCFG Write Address Misalignment received by UBOX
4:4	RWS_V	0x0	MMCFG Read Address Misalignment received by UBOX(CFGRdAddrMisAligned): MMCFG Read Address Misalignment received by UBOX
3:3	RWS_V	0x0	Unsupported Opcode received by UBOX(UnsupportedOpcode): Unsupported opcode received by UBOX
2:2	RWS_V	0x0	Poison was received by UBOX(PoisonRsvd): UBOX received a poisoned transaction
1:1	RWS_V	0x0	SMI source iMC(SMISrciMC): SMI is caused due to an indication from the iMC
0:0	RWS_V	0x0	SMI is caused due to a locally generated UMC(SMISrcUMC): This is a bit that indicates that an SMI was caused due to a locally generated UMC

5.2 Device 11 Function 2

DID		VID		0h	80h
PCISTS		PCICMD		4h	84h
CCR			RID	8h	88h
BIST	HDR	PLAT	CLSR	Ch	8Ch
				10h	90h
				14h	94h
				18h	98h
				1Ch	9Ch
				20h	A0h
				24h	A4h
				28h	A8h
SDID		SVID		2Ch	ACh
				30h	B0h
			CAPPTR	34h	B4h
				38h	B8h
MAXLAT	MINGNT	INTPIN	INTL	3Ch	BCh
				40h	C0h
				44h	C4h

5.3 Device 11 Function 3

Table 5-2. UBOX Device 11 Function 3 Register Address Map

Register Name	Offset	Size
CPUBUSNO	0xd0	32
SMICtrl	0xd8	32

5.3.1 CPUBUSNO

Bus Number Configuration for the processor.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**11**	**Function:**	**3**
Offset:	**0xd0**				

Bit	Attr	Default	Description
31:31	RW-LB	0x0	Valid: Indicates whether the bus numbers have been initialized or not
15:8	RW-LB	0x0	The processor Bus Number 1(CPUBUSNO1): Bus Number for non IIO devices in the Uncore
7:0	RW-LB	0x0	The processor Bus Number 0(CPUBUSNO0): Bus Number for IIO devices

5.3.2 SMICtrl

SMI generation control

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**1**	**Device:**	**11**	**Function:**	**3**
Offset:	**0xd8**				

Bit	Attr	Default	Description
27:27	RW-LB	0x0	Disable Generation of SMI for new Ubox erros (SMIDis3): Disable generation of SMI from message channel
26:26	RW-LB	0x1	Disable Generation of SMI for new Ubox erros (SMIDis2): Disable generation of SMI for new Ubox errors
25:25	RW-LB	0x0	Disable Generation of SMI (all) (SMIDis): Disable generation of SMI
24:24	RW-LB	0x0	UMC SMI Enable (UMCSMIEn): This is the enable bit that enables SMI generation due to a UMC 1 - Generate SMI after the threshold counter expires. 0 - Disable generation of SMI
19:0	RW-LB	0x0	SMI generation threshold (Threshold): This is the countdown that happens in the hardware before an SMI is generated due to a UMC.

§

6 Power Controller Unit (PCU) Register

6.1 Device 10 Function 0

Table 6-1. PCU Device 10 Function 0 Register Address Map

Register Name	Offset	Size
MEM_TRML_TEMPERATURE_REPORT	0x60	32
MEM_ACCUMULATED_BW_CH_0	0x64	32
MEM_ACCUMULATED_BW_CH_1	0x68	32
MEM_ACCUMULATED_BW_CH_2	0x6c	32
MEM_ACCUMULATED_BW_CH_3	0x70	32
PACKAGE_POWER_SKU	0x84	64
PACKAGE_POWER_SKU_UNIT	0x8c	32
PACKAGE_ENERGY_STATUS	0x90	32
Package_Temperature	0xc8	32
P_State_Limits	0xd8	32
TEMPERATURE_TARGET	0xe4	32

6.1.1 MEM_TRML_TEMPERATURE_REPORT

This register is used to report the thermal status of the memory.

The channel maximum temperature field is used to report the maximal temperature of all ranks.

THIS REGISTER IS DUPLICATED IN THE PCU I/O SPACE, CHANGES MUST BE MADE IN BOTH PLACES.

Type: CFG **Bus:** 1 **Offset:** 0x60			**PortID:** N/A **Device:** 10 **Function:**0
Bit	**Attr**	**Default**	**Description**
31:24	RO-V	0x0	Channel 3 Maximum Temperature (Channel3_Max_Temperature): Temperature in Degrees (C).
23:16	RO-V	0x0	Channel 2 Maximum Temperature (Channel2_Max_Temperature): Temperature in Degrees (C).
15:8	RO-V	0x0	Channel 1 Maximum Temperature (Channel1_Max_Temperature): Temperature in Degrees (C).
7:0	RO-V	0x0	Channel 0 Maximum Temperature (Channel0_Max_Temperature): Temperature in Degrees (C).

6.1.2 MEM_ACCUMULATED_BW_CH_[0:3]

This register contains a measurement proportional to the weighted DRAM BW for the channel including all ranks. The weights are configured in the memory controller channel register PM_CMD_PWR.

THIS REGISTER IS DUPLICATED IN THE PCU I/O SPACE, CHANGES MUST BE MADE IN BOTH PLACES.

Type: **Bus:** **Offset:**	**CFG** **1** **0x64, 0x68, 0x6c, 0x70**	**PortID: N/A** **Device: 10**	 **Function:0**
Bit	**Attr**	**Default**	**Description**
31:0	RO-V	0x0	Data (DATA): The weighted BW value is calculated by the memory controller based on the following formula: NumPrecharge * PM_CMD_PWR[PWR_RAS_PRE] + NumReads * PM_CMD_PWR[PWR_CAS_R] + NumWrites * PM_CMD_PWR[PWR_CAS_W]

6.1.3 PACKAGE_POWER_SKU

Defines allowed SKU power and timing parameters.

Type: **Bus:** **Offset:**	**CFG** **1** **0x84**	**PortID: N/A** **Device: 10**	 **Function:0**
Bit	**Attr**	**Default**	**Description**
54:48	RO-V	0x18	Maximal Time Window (PKG_MAX_WIN): The maximal time window allowed for the SKU. Higher values will be clamped to this value. The timing interval window is Floating Point number given by power (2,PKG_MAX_WIN). The unit of measurement is defined in PACKAGE_POWER_SKU_UNIT_MSR[TIME_UNIT].
46:32	RO-V	0x258	Maximal Package Power (PKG_MAX_PWR): The maximal package power setting allowed for the SKU. Higher values will be clamped to this value. The maximum setting is typical not guaranteed. The units for this value are defined in PACKAGE_POWER_SKU_UNIT_MSR[PWR_UNIT].
30:16	RO-V	0x78	Minimal Package Power (PKG_MIN_PWR): The minimal package power setting allowed for the SKU. Lower values will be clamped to this value. The minimum setting is typical not guaranteed. The units for this value are defined in PACKAGE_POWER_SKU_UNIT_MSR[PWR_UNIT].
14:0	RO-V	0x118	TDP Package Power (PKG_TDP): The TDP package power setting allowed for the SKU. The TDP setting is typical not guaranteed. The units for this value are defined in PACKAGE_POWER_SKU_UNIT_MSR[PWR_UNIT].

6.1.4 PACKAGE_POWER_SKU_UNIT

Defines units for calculating SKU power and timing parameters.

PCODE will update the contents of this register.

Type:	**CFG**	**PortID:**	**N/A**	
Bus:	**1**	**Device:**	**10**	**Function:0**
Offset:	**0x8c**			

Bit	Attr	Default	Description
19:16	RO-V	0xa	Time Unit (TIME_UNIT): Time Units used for power control registers. The actual unit value is calculated by 1/Power(2,TIME_UNIT) second. The default value of 0Ah corresponds to 976 usec.
12:8	RO-V	0x10	Energy Units (ENERGY_UNIT): Energy Units used for power control registers. The actual unit value is calculated by 1/Power(2,ENERGY_UNIT) J. The default value of 10h corresponds to 15.3 uJ.
3:0	RO-V	0x3	Power Units (PWR_UNIT): Power Units used for power control registers. The actual unit value is calculated by 1/Power(2,PWR_UNIT) W. The default value of 0011b corresponds to 18 W.

6.1.5 PACKAGE_ENERGY_STATUS

Package energy consumed by the entire PCODE including IA, GT, and Uncore. The counter will wrap around and continue counting when it reaches its limit.

The energy status is reported in units which are defined in PACKAGE_POWER_SKU_UNIT_MSR[ENERGY_UNIT].

The data is updated by PCODE and is Read Only for all software.

THIS REGISTER IS DUPLICATED IN THE PCU I/O SPACE, CHANGES MUST BE MADE IN BOTH PLACES.

Type:	**CFG**	**PortID:**	**N/A**	
Bus:	**1**	**Device:**	**10**	**Function:0**
Offset:	**0x90**			

Bit	Attr	Default	Description
31:0	RO-V	0x0	Energy Value (DATA): Energy Value

6.1.6 Package_Temperature

Package temperature in degrees (C). This field is updated by firmware.

THIS REGISTER IS DUPLICATED IN THE PCU I/O SPACE, CHANGES MUST BE MADE IN BOTH PLACES.

Type:	**CFG**	**PortID:**	**N/A**	
Bus:	**1**	**Device:**	**10**	**Function:0**
Offset:	**0xc8**			

Bit	Attr	Default	Description
7:0	RO-V	0x0	Temperature (DATA): Package temperature in degrees (C).

6.1.7 P_State_Limits

This register allows software to limit the maximum frequency allowed during run-time.

PCODE will sample this register in slow loop.

Functionality added in B-step.

Type:	**CFG**	**PortID:**	**N/A**	
Bus:	**1**	**Device:**	**10**	**Function:0**
Offset:	**0xd8**			

Bit	Attr	Default	Description
31:31	RW_KL	0x0	Lock (LOCK): This bit will lock all settings in this register.
15:8	RW-L	0x0	P-State Offset (PSTT_OFFSET): Hardware P-State control on the relative offset from P1. The offset field determines the number of bins to drop P1 (dynamically).
7:0	RW-L	0xff	P-State Limitation (PSTT_LIM): This field indicates the maximum frequency limit allowed during run-time.

6.1.8 TEMPERATURE_TARGET

Legacy register holding temperature related constants for Platform use. This register is updated by firmware.

Type:	**CFG**	**PortID:**	**N/A**	
Bus:	**1**	**Device:**	**10**	**Function:0**
Offset:	**0xe4**			

Bit	Attr	Default	Description
27:24	RO-V	0x0	TJ_{MAX} TCC Offset (TJ_MAX_TCC_OFFSET): Temperature offset in degrees (C) from the TJ_{MAX}. Used for throttling temperature. Will not impact temperature reading. If offset is allowed and set - the throttle will occur and reported at lower than Tj_{MAX}

Type: CFG **PortID:** N/A
Bus: 1 **Device:** 10 **Function:**0
Offset: 0xe4

Bit	Attr	Default	Description
23:16	RO-V	0x0	Thermal Monitor Reference Temperature (REF_TEMP): This field indicates the maximum junction temperature, also referred to as the throttle temperature, TCC activation temperature or prochot temperature. This is the temperature at which the Thermal Monitor is activated. Firmware will update this register with the following value: 125 minus FUSETJMAXOFFSET.
15:8	RO-V	0x0	Fan Temperature target offset (FAN_TEMP_TARGET_OFST): Fan Temperature target offset Also Known As T-Control. Indicates the relative offset from the Thermal Monitor Trip Temperature at which fans should be engaged.

§

7 Integrated I/O (IIO) Configuration Registers

7.1 Registers Overview

7.1.1 Configuration Registers (CSR)

There are two distinct CSR register spaces supported by the IIO Module.

The first one is the traditional PCI-defined configuration registers. These registers are accessed by means of the well known configuration transaction mechanism defined in the PCI specification and this uses the bus:device:function number concept to address a specific device’s configuration space.

The second is by means of MMIO space for Intel VT-d, RCRB and I/OxAPIC runtime registers.

7.1.2 BDF:BAR# for Various MMIO BARs in IIO

This is needed for any entity trying to access MMIO registers in the IIO module over message channel.

Table 7-1. BDF:BAR# for Various MMIO BARs in IIO

BAR Name	B	D	F	BAR#
DMIRCBAR	DC	0	0	0
CB-BAR0	DC	4	0	0
CB-BAR1	DC	4	1	0
CB-BAR2	DC	4	2	0
CB-BAR3	DC	4	3	0
CB-BAR4	DC	4	4	0
CB-BAR5	DC	4	5	0
CB-BAR6	DC	4	6	0
CB-BAR7	DC	4	7	0
VT-d VTBAR	DC	5	0	0
I/OxAPIC-MBAR	DC	5	4	0
I/OxAPIC-ABAR	DC	5	4	1

7.1.3 Unimplemented Devices / Functions and Registers

If the IIO module receives a configuration access over message channel or directly by means of the JTAG mini-port, to a device/function or BAR# that does not exist in the IIO module, the IIO module will abort these accesses. Software should not attempt or rely on reads or writes to unimplemented registers or register bits.

7.1.4 PCI versus PCIe Device/Function

PCI devices/functions do NOT have a PCIe capability register set and do not decode offsets 100h and beyond. Accesses to 100h and beyond are master aborted by these devices. I/OxAPIC functions are PCI functions. All other functions in the IIO module are PCIe functions and these have a PCIe capability register set and also decode address offsets 100h and beyond.

7.2 Device 0 Function 0 DMI, Device 0 Function 0 PCIe, Device 1 Function 0-1 PCIe, Device 2 Function 0-3 PCIe, Device 3 Function 0-3 PCIe

Device 0 Function 0 PCIe Mode - Port 0 (X4)

Device 1 - Port 1 (X8)

Device 2 - Port 2 (X16)

Device 3 - Port 3 (X16)

Table 7-2. Function Number of Active Root Ports in Port 1 (Device 1) Based on Port Bifurcation

Port Bifurcation	Function# of Active Root Port			
			7:4	3:0
x8			0	
x4x4			1	0

Table 7-3. Function Number of Active Root Ports in Port 2 (Device 2) Based on Port Bifurcation

Port Bifurcation	Function# of Active Root Port			
	15:12	11:8	7:4	3:0
x16	0			
x8x8	2		0	
x8x4x4	2		1	0
x4x4x8	3	2	0	
x4x4x4x4	3	2	1	0

Table 7-4. Function Number of Active Root Ports in Port 3 (Device 3) Based on Port Bifurcation

Port Bifurcation	Function# of Active Root Port			
	15:12	11:8	7:4	3:0
x16	0			
x8x8	2		0	
x8x4x4	2		1	0
x4x4x8	3	2	0	
x4x4x4x4	3	2	1	0

Table 7-5. Integrated I/O Register Address Map (Sheet 1 of 4)

Register Name	Offset	Size	Device 0 Function	Device 1 Function	Device 2 Function	Device 3 Function
vid	0x0	16	0	0-1	0 - 3	0 - 3
did	0x2	16	0	0-1	0 - 3	0 - 3
pcicmd	0x4	16	0	0-1	0 - 3	0 - 3
pcists	0x6	16	0	0-1	0 - 3	0 - 3
rid	0x8	8	0	0-1	0 - 3	0 - 3
ccr	0x9	24	0	0-1	0 - 3	0 - 3
clsr	0xc	8	0	0-1	0 - 3	0 - 3
plat	0xd	8	0	0-1	0 - 3	0 - 3
hdr	0xe	8	0	0-1	0 - 3	0 - 3
bist	0xf	8	0	0-1	0 - 3	0 - 3
pbus	0x18	8	0 (PCIe)	0-1	0 - 3	0 - 3
secbus	0x19	8	0 (PCIe)	0-1	0 - 3	0 - 3
subbus	0x1a	8	0 (PCIe)	0-1	0 - 3	0 - 3
iobas	0x1c	8	0 (PCIe)	0-1	0 - 3	0 - 3
iolim	0x1d	8	0 (PCIe)	0-1	0 - 3	0 - 3
secsts	0x1e	16	0 (PCIe)	0-1	0 - 3	0 - 3
mbas	0x20	16	0 (PCIe)	0-1	0 - 3	0 - 3
mlim	0x22	16	0 (PCIe)	0-1	0 - 3	0 - 3
pbas	0x24	16	0 (PCIe)	0-1	0 - 3	0 - 3
plim	0x26	16	0 (PCIe)	0-1	0 - 3	0 - 3
pbasu	0x28	32	0 (PCIe)	0-1	0 - 3	0 - 3
plimu	0x2c	32	0 (PCIe)	0-1	0 - 3	0 - 3
capptr	0x34	8	0	0-1	0 - 3	0 - 3
intl	0x3c	8	0	0-1	0 - 3	0 - 3
intpin	0x3d	8	0	0-1	0 - 3	0 - 3
bctrl	0x3e	16	0 (PCIe)	0-1	0 - 3	0 - 3
scapid	0x40	8	0 (PCIe)	0-1	0 - 3	0 - 3
snxtptr	0x41	8	0 (PCIe)	0-1	0 - 3	0 - 3
svid	0x2c	16	0 (DMI2)			
svid	0x44	16	0 (PCIe)	0-1	0 - 3	0 - 3
sdid	0x2e	16	0 (DMI2)			
sdid	0x46	16	0 (PCIe)	0-1	0 - 3	0 - 3
dmircbar	0x50	32	0	0-1		
msicapid	0x60	8	0	0-1	0 - 3	0 - 3
msinxtptr	0x61	8	0	0-1	0 - 3	0 - 3
msimsgctl	0x62	16	0	0-1	0 - 3	0 - 3
msgadr	0x64	32	0	0-1	0 - 3	0 - 3
msgdat	0x68	32	0	0-1	0 - 3	0 - 3
msimsk	0x6c	32	0	0-1	0 - 3	0 - 3
msipending	0x70	32	0	0-1	0 - 3	0 - 3
pxpcapid	0x90	8	0	0-1	0 - 3	0 - 3
pxpnxtptr	0x91	8	0	0-1	0 - 3	0 - 3

Table 7-5. Integrated I/O Register Address Map (Sheet 2 of 4)

Register Name	Offset	Size	Device 0 Function	Device 1 Function	Device 2 Function	Device 3 Function
pxpcap	0x92	16	0	0-1	0 - 3	0 - 3
devcap	0x94	32	0	0-1	0 - 3	0 - 3
devctrl	0xf0	16	0 (DMI2)			
devctrl	0x98	16	0 (PCIe)	0-1	0 - 3	0 - 3
devsts	0xf2	16	0 (DMI2)			
devsts	0x9a	16	0 (PCIe)	0-1	0 - 3	0 - 3
lnkcap	0x9c	32	0	0-1	0 - 3	0 - 3
lnkcon	0x1b0	16	0 (DMI2)			
lnkcon	0xa0	16	0 (PCIe)	0-1	0 - 3	0 - 3
lnksts	0x1b2	16	0 (DMI2)			
lnksts	0xa2	16	0 (PCIe)	0-1	0 - 3	0 - 3
sltcap	0xa4	32	0 (PCIe)	0-1	0 - 3	0 - 3
sltcon	0xa8	16	0 (PCIe)	0-1	0 - 3	0 - 3
sltsts	0xaa	16	0 (PCIe)	0-1	0 - 3	0 - 3
rootcon	0xac	16	0	0-1	0 - 3	0 - 3
rootcap	0xae	16	0	0-1	0 - 3	0 - 3
rootsts	0xb0	32	0 (PCIe)	0-1	0 - 3	0 - 3
devcap2	0xb4	32	0	0-1	0 - 3	0 - 3
devctrl2	0xf8	16	0 (DMI2)			
devctrl2	0xb8	16	0 (PCIe)	0-1	0 - 3	0 - 3
lnkcap2	0xbc	32	0	0-1	0 - 3	0 - 3
lnkcon2	0x1c0	16	0 (DMI2)			
lnkcon2	0xc0	16	0 (PCIe)	0-1	0 - 3	0 - 3
lnksts2	0x1c2	16	0 (DMI2)			
lnksts2	0xc2	16	0 (PCIe)	0-1	0 - 3	0 - 3
pmcap	0xe0	32	0	0-1	0 - 3	0 - 3
pmcsr	0xe4	32	0	0-1	0 - 3	0 - 3
xpreut_hdr_ext	0x100	32	0	0-1	0 - 3	0 - 3
xpreut_hdr_cap	0x104	32	0	0-1	0 - 3	0 - 3
xpreut_hdr_lef	0x108	32	0	0-1	0 - 3	0 - 3
acscaphdr	0x110	32	0 (PCIe)	0-1	0 - 3	0 - 3
acscap	0x114	16	0 (PCIe)	0-1	0 - 3	0 - 3
acsctrl	0x116	16	0 (PCIe)	0-1	0 - 3	0 - 3
apicbase	0x140	16	0	0-1	0 - 3	0 - 3
apiclimit	0x142	16	0	0-1	0 - 3	0 - 3
vsecphdr	0x144	32	0 (DMI2)			
vshdr	0x148	32	0 (DMI2)			
errcaphdr	0x148	32	0 (PCIe)	0-1	0 - 3	0 - 3
uncerrsts	0x14c	32	0	0-1	0 - 3	0 - 3
uncerrmsk	0x150	32	0	0-1	0 - 3	0 - 3
uncerrsev	0x154	32	0	0-1	0 - 3	0 - 3
corerrsts	0x158	32	0	0-1	0 - 3	0 - 3
corerrmsk	0x15c	32	0	0-1	0 - 3	0 - 3

Table 7-5. Integrated I/O Register Address Map (Sheet 3 of 4)

Register Name	Offset	Size	Device 0 Function	Device 1 Function	Device 2 Function	Device 3 Function
errcap	0x160	32	0	0-1	0 - 3	0 - 3
hdrlog0	0x164	32	0	0-1	0 - 3	0 - 3
hdrlog1	0x168	32	0	0-1	0 - 3	0 - 3
hdrlog2	0x16c	32	0	0-1	0 - 3	0 - 3
hdrlog3	0x170	32	0	0-1	0 - 3	0 - 3
rperrcmd	0x174	32	0	0-1	0 - 3	0 - 3
rperrsts	0x178	32	0	0-1	0 - 3	0 - 3
errsid	0x17c	32	0	0-1	0 - 3	0 - 3
perfctrlsts_0	0x180	32	0	0-1	0 - 3	0 - 3
perfctrlsts_1	0x184	32	0	0-1	0 - 3	0 - 3
miscctrlsts_0	0x188	32	0	0-1	0 - 3	0 - 3
miscctrlsts_1	0x18c	32	0	0-1	0 - 3	0 - 3
pcie_iou_bif_ctrl	0x190	16	0	0-1	0	0
dmictrl	0x1a0	64	0 (DMI2)			
dmists	0x1a8	32	0 (DMI2)			
ERRINJCAP	0x1d0	32	0	0-1	0 - 3	0 - 3
ERRINJHDR	0x1d4	32	0	0-1	0 - 3	0 - 3
ERRINJCON	0x1d8	16	0	0-1	0 - 3	0 - 3
ctoctrl	0x1e0	32	0	0-1	0 - 3	0 - 3
xpcorerrsts	0x200	32	0	0-1	0 - 3	0 - 3
xpcorerrmsk	0x204	32	0	0-1	0 - 3	0 - 3
xpuncerrsts	0x208	32	0	0-1	0 - 3	0 - 3
xpuncerrmsk	0x20c	32	0	0-1	0 - 3	0 - 3
xpuncerrsev	0x210	32	0	0-1	0 - 3	0 - 3
xpuncerrptr	0x214	8	0	0-1	0 - 3	0 - 3
uncedmask	0x218	32	0	0-1	0 - 3	0 - 3
coredmask	0x21c	32	0	0-1	0 - 3	0 - 3
rpedmask	0x220	32	0	0-1	0 - 3	0 - 3
xpuncedmask	0x224	32	0	0-1	0 - 3	0 - 3
xpcoredmask	0x228	32	0	0-1	0 - 3	0 - 3
xpglberrsts	0x230	16	0	0-1	0 - 3	0 - 3
xpglberrptr	0x232	16	0	0-1	0 - 3	0 - 3
pxp2cap	0x250	32		0-1	0 - 3	0 - 3
lnkcon3	0x254	32		0-1	0 - 3	0 - 3
lnerrsts	0x258	32		0-1	0 - 3	0 - 3
ln0eq	0x25c	16		0-1	0 - 3	0 - 3
ln1eq	0x25e	16		0-1	0 - 3	0 - 3
ln2eq	0x260	16		0-1	0 - 3	0 - 3
ln3eq	0x262	16		0-1	0 - 3	0 - 3
ln4eq	0x264	16		0-1	0, 2	0, 2
ln5eq	0x266	16		0-1	0, 2	0, 2
ln6eq	0x268	16		0-1	0, 2	0, 2
ln7eq	0x26a	16		0-1	0, 2	0, 2

Table 7-5. Integrated I/O Register Address Map (Sheet 4 of 4)

Register Name	Offset	Size	Device 0 Function	Device 1 Function	Device 2 Function	Device 3 Function
ln8eq	0x26c	16			0	0
ln9eq	0x26e	16			0	0
ln10eq	0x270	16			0	0
ln11eq	0x272	16			0	0
ln12eq	0x274	16			0	0
ln13eq	0x276	16			0	0
ln14eq	0x278	16			0	0
ln15eq	0x27a	16			0	0
ler_cap	0x280	32	0	0-1	0 - 3	0 - 3
ler_hdr	0x284	32	0	0-1	0 - 3	0 - 3
ler_ctrlsts	0x288	32	0	0-1	0 - 3	0 - 3
ler_uncerrmsk	0x28c	32	0	0-1	0 - 3	0 - 3
ler_xpuncerrmsk	0x290	32	0	0-1	0 - 3	0 - 3
ler_rperrmsk	0x294	32	0	0-1	0 - 3	0 - 3
xppmdfxmat0	0x2f0	32	0	0	0	0
xppmdfxmat1	0x2f4	32	0	0	0	0
xppmdfxmsk0	0x2f8	32	0	0	0	0
xppmdfxmsk1	0x2fc	32	0	0	0	0
xppmdl0	0x480	32	0	0	0	0
xppmdl1	0x484	32	0	0	0	0
xppmcl0	0x488	32	0	0	0	0
xppmcl1	0x48c	32	0	0	0	0
xppmdh	0x490	16	0	0	0	0
xppmch	0x492	16	0	0	0	0
xppmr0	0x494	32	0	0	0	0
xppmr1	0x498	32	0	0	0	0
xppmevl0	0x49c	32	0	0	0	0
xppmevl1	0x4a0	32	0	0	0	0
xppmevh0	0x4a4	32	0	0	0	0
xppmevh1	0x4a8	32	0	0	0	0
xppmer0	0x4ac	32	0	0	0	0
xppmer1	0x4b0	32	0	0	0	0

7.2.1 vid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x0**				

Bit	Attr	Default	Description
15:0	RO	0x8086	vendor_identification_number: The value is assigned by PCI-SIG to Intel.

7.2.2 did

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x2**				

Bit	Attr	Default	Description
15:0	RO RO-V (Device 0 and 3 Function 0)	For Device 0 Function 0: 0xe00 (DMI2 Mode) 0xe01 (PCIe Mode) For Device 2: 0xe04 (Function 0) 0xe05 (Function 1) 0xe06 (Function 2) 0xe07 (Function 3) For Device 3: 0xe08 (Function 0) 0xe09 (Function 1) 0xe0a (Function 2) 0xe0b (Function 3)	device_identification_number: Device ID values vary from function to function. Bits 15:8 are equal to 0x0E .

7.2.3 pcicmd

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x4**				

Bit	Attr	Default	Description
10:10	RW	0x0	interrupt_disable: Interrupt Disable. Controls the ability of the PCI Express* port to generate INTx messages. This bit does not affect the ability of the processor to route interrupt messages received at the PCI Express* port. However, this bit controls the generation of legacy interrupts to the DMI for PCI Express* errors detected internally in this port (for example, Malformed TLP, CRC error, completion time out, and so forth) or when receiving RP error messages or interrupts due to HP/PM events generated in legacy mode within the processor. 1: Legacy Interrupt mode is disabled 0: Legacy Interrupt mode is enabled
9:9	RO	0x0	fast_back_to_back_enable: Fast Back-to-Back Enable Not applicable to PCI Express* must be hardwired to 0.
8:8	RW	0x0	serre: SERR Enable For PCI Express*/DMI ports, this field enables notifying the internal core error logic of occurrence of an uncorrectable error (fatal or non-fatal) at the port. The internal core error logic of the IIO module then decides if/how to escalate the error further (pins/ message, and so forth). This bit also controls the propagation of PCI Express* ERR_FATAL and ERR_NONFATAL messages received from the port to the internal IIO core error logic. 1: Fatal and Non-fatal error generation and Fatal and Non-fatal error message forwarding is enabled 0: Fatal and Non-fatal error generation and Fatal and Non-fatal error message forwarding is disabled Refer to PCI Express* Base Specification, Revision 2.0 for details of how this bit is used in conjunction with other control bits in the Root Control register for forwarding errors detected on the PCI Express* interface to the system core error logic.
7:7	RO	0x0	idsel_stepping_wait_cycle_control: IDSEL Stepping/Wait Cycle Control Not applicable to PCI Express* must be hardwired to 0.
6:6	RW	0x0	perre: Parity Error Response For PCI Express*/DMI ports, the IIO module ignores this bit and always does parity checking and signaling for data/address of transactions both to and from IIO. This bit though affects the setting of bit 8 in the PCISTS register.
5:5	RO	0x0	vga_palette_snoop_enable: Not applicable to PCI Express* must be hardwired to 0.
4:4	RO	0x0	mwie: Not applicable to PCI Express* must be hardwired to 0.
3:3	RO	0x0	sce: Not applicable to PCI Express* must be hardwired to 0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x4**				

Bit	Attr	Default	Description
2:2	RW RW-L (Device 0 Function 0)	0x0	bme:
1:1	RW RW-L (Device 0 Function 0)	0x0	mse: Memory Space Enable 1: Enables a PCI Express* port's memory range registers to be decoded as valid target addresses for transactions from secondary side. 0: Disables a PCI Express* port's memory range registers (including the Configuration Registers range registers) to be decoded as valid target addresses for transactions from secondary side. All memory accesses received from secondary side are UR'ed.
0:0	RW RW-L (Device 0 and 3 Function 0)	0x0	iose: I/O Space Enable Controls a device's response to I/O Space accesses. A value of 0 disables the device response. A value of 1 allows the device to respond to I/O Space accesses. State after RST# is 0.

7.2.4 pcists

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	0	Function:	0
Bus:	0	Device:	2	Function:	0-3
Bus:	0	Device:	3	Function:	0-3
Offset:	0x6				

Bit	Attr	Default	Description
15:15	RW1C	0x0	dpe: Detected Parity Error This bit is set by a root port when it receives a packet on the primary side with an uncorrectable data error (including a packet with poison bit set) or an uncorrectable address/control parity error. The setting of this bit is regardless of the Parity Error Response bit (PERRE) in the PCICMD register.
14:14	RW1C	0x0	sse: Signaled System Error 1: The root port reported fatal/non-fatal (and not correctable) errors it detected on its PCI Express* interface to the IIO core error logic (which might eventually escalate the error through the ERR[2:0] pins or message to processor core or message to PCH). **Note:** The SERRE bit in the PCICMD register must be set for a device to report the error the IIO core error logic.Software clears this bit by writing a '1' to it. This bit is also set (when SERR enable bit is set) when a FATAL/NON-FATAL message is forwarded to the IIO core error logic. **Note:** The IIO internal 'core' errors (like parity error in the internal queues) are not reported by means of this bit. 0: The root port did not report a fatal/non-fatal error
13:13	RW1C	0x0	rma: Received Master Abort This bit is set when a root port experiences a master abort condition on a transaction it mastered on the primary interface (Uncore internal bus). **Note**: Certain errors might be detected right at the PCI Express* interface and those transactions might not 'propagate' to the primary interface before the error is detected (for example, accesses to memory above TOCM in cases where the PCIe interface logic itself might have visibility into TOCM). Such errors do not cause this bit to be set, and are reported by means of the PCI Express* interface error bits (secondary status register). Conditions that cause bit 13 to be set, include: Device receives a completion on the primary interface (internal bus of Uncore) with Unsupported Request or master abort completion Status. This includes UR status received on the primary side of a PCI Express* port on peer-to-peer completions also.
12:12	RW1C	0x0	rta: Received Target Abort This bit is set when a device experiences a completer abort condition on a transaction it mastered on the primary interface (Uncore internal bus). **Note**: Certain errors might be detected right at the PCI Express* interface and those transactions might not 'propagate' to the primary interface before the error is detected (for example, accesses to memory above VTBAR). Such errors do not cause this bit to be set, and are reported by means of the PCI Express* interface error bits (secondary status register). Conditions that cause bit 12 to be set, include: Device receives a completion on the primary interface (internal bus of Uncore) with completer abort completion Status. This includes CA status received on the primary side of a PCI Express* port on peer-to-peer completions also. Other completer abort conditions detected on the Uncore internal bus.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x6**				

Bit	Attr	Default	Description
11:11	RW1C	0x0	sta: Signaled Target Abort This bit is set when a root port signals a completer abort completion status on the primary side (internal bus of Uncore). This condition includes a PCI Express* port forwarding a completer abort status received on a completion from the secondary.
10:9	RO	0x0	devsel_timing: Not applicable to PCI Express*. Hardwired to 0.
8:8	RW1C	0x0	mdpe: Master Data Parity Error This bit is set by a root port if the Parity Error Response bit in the PCI Command register is set and it either receives a completion with poisoned data from the primary side or it forwards a packet with data (including MSI writes) to the primary side with poison.
7:7	RO	0x0	fast_back_to_back: Not applicable to PCI Express*. Hardwired to 0.
5:5	RO	0x0	pci66mhz_capable: Not applicable to PCI Express*. Hardwired to 0.
4:4	RO	0x1	capabilities_list: Not applicable to PCI Express*. Hardwired to 0.
3:3	RO-V	0x0	intx_status: This Read-only bit reflects the state of the interrupt in the PCI Express* Root Port. Only when the Interrupt Disable bit in the command register is a 0 and this Interrupt Status bit is a 1, will this device generate INTx interrupt. Setting the Interrupt Disable bit to a 1 has no effect on the state of this bit.This bit does not get set for interrupts forwarded to the root port from downstream devices in the hierarchy. When MSI are enabled, Interrupt status should not be set.

7.2.5 rid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x8**				

Bit	Attr	Default	Description
7:0	RO-V	0x0	revision_id: Reflects the Uncore Revision ID after reset. Reflects the Compatibility Revision ID after the BIOS writes 0x69 to any RID register in any processor function. **Implementation Note:** Read and write requests from the host to any RID register in any processor function are re-directed to the IIO cluster. Accesses to the CCR field are also redirected due to Dword alignment. It is possible that JTAG accesses are direct, so will not always be redirected.

7.2.6 ccr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x9**				

Bit	Attr	Default	Description
23:16	RO-V	0x6	base_class: Generic Device
15:8	RO-V	0x4 0x80 (Device 3 Function 0 only)	sub_class: Generic Device
7:0	RO-V	0x0	interface: This field is hardwired to 00h for PCI Express* port.

7.2.7 clsr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xc**				

Bit	Attr	Default	Description
7:0	RW	0x0	cacheline_size: This register is set as RW for compatibility reasons only. Cacheline size is always 64B. IIO hardware ignores this setting.

7.2.8 plat

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xd**				

Bit	Attr	Default	Description
7:0	RO	0x0	primary_latency_timer: Not applicable to PCI-Express*. Hardwired to 00h.

7.2.9 hdr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xe**				

Bit	Attr	Default	Description
7:7	RO-V RO (Device 0 Function 0)	0x1 0x0 (Device 0 Function 0)	mfd: Multi-function Device This bit defaults to 0 for Device 0. This bit defaults to 1 for Devices 2-3. The BIOS can individually control the value of this bit in Function 0 of these devices, based on HDRTYPCTRL register. The BIOS will write to that register to change this field to 0 in Function 0 of these devices, if it exposes only Function 0 in the device to OS. **Note:** In product SKUs where only Function 0 of the device is exposed to any software (BIOS/OS), the BIOS would have to still set the control bits mentioned above to set the this bit in this register to be compliant per PCI rules.
6:0	RO RO-V (Device 0 Function 0)	0x1 0x0 (Device 0 Function 0)	cl: Configuration Layout This field identifies the format of the configuration header layout. In DMI mode, default is 00h indicating a conventional type 00h PCI header. In PCIe mode, the default is 01h, corresponding to Type 1 for a PCIe root port.

7.2.10 bist

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xf**				

Bit	Attr	Default	Description
7:0	RO	0x0	bist_tests: Not Supported. Hardwire to 00h.

7.2.11 pbus

Primary Bus Number Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x18**				

Bit	Attr	Default	Description
7:0	RW	0x0	pbn: Configuration software programs this field with the number of the bus on the primary side of the bridge. This register has to be kept consistent with the Internal Bus Number 0 in the CPUBUSNO01 register. The BIOS (and OS if internal bus number gets moved) must program this register to the correct value since IIO hardware would depend on this register for inbound configuration cycle decode purposes.

7.2.12 secbus

Secondary Bus Number Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x19**				

Bit	Attr	Default	Description
7:0	RW	0x0	sbn: This field is programmed by configuration software to assign a bus number to the secondary bus of the virtual P2P bridge. IIO uses this register to either forward a configuration transaction as a Type 1 or Type 0 to PCI Express*.

7.2.13 subbus

Subordinate Bus Number Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1a**				

Bit	Attr	Default	Description
7:0	RW	0x0	subordinate_bus_number: This register is programmed by configuration software with the number of the highest subordinate bus that is behind the PCI Express* port. Any transaction that falls between the secondary and subordinate bus number (both inclusive) of an Express* port is forwarded to the Express* port.

7.2.14 iobas

I/O Base Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1c**				

Bit	Attr	Default	Description
7:4	RW	0xf	i_o_base_address: Corresponds to A[15:12] of the I/O base address of the PCI Express* port. See also the IOLIM register description.
3:2	RW-L	0x0	more_i_o_base_address: When EN1K is set in the IIOMISCCTRL register, these bits become RW and allow for 1K granularity of I/O addressing, otherwise these are RO.
1:0	RO	0x0	i_o_address_capability: IIO supports only 16-bit addressing

7.2.15 iolim

I/O Limit Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1d**				

Bit	Attr	Default	Description
7:4	RW	0x0	i_o_address_limit: Corresponds to A[15:12] of the I/O limit address of the PCI Express* port.The I/O Base and I/O Limit registers define an address range that is used by the PCI Express* port to determine when to forward I/O transactions from one interface to the other using the following formula: IO_BASE <= A[15:12] <= IO_LIMIT The bottom of the defined I/O address range will be aligned to a 4KB boundary (1KB if EN1K bit is set. Refer to the IIOMISCCTRL register for definition of EN1K bit) while the top of the region specified by IO_LIMIT will be one less than a 4KB (1KB if EN1K bit is set) multiple. **Notes:** • Setting the I/O limit less than I/O base disables the I/O range altogether. • General the I/O base and limit registers won't be programmed by software without clearing the IOSE bit first.
3:2	RW-L	0x0	more_i_o_address_limit: When EN1K is set in the IIOMISCCTRL register, these bits become RW and allow for 1K granularity of I/O addressing, otherwise these are RO.
1:0	RO	0x0	i_o_address_limit_capability: IIO only supports 16-bit addressing

7.2.16 secsts

Secondary Status Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1e**				

Bit	Attr	Default	Description
15:15	RW1C	0x0	dpe: Detected Parity Error This bit is set by the root port whenever it receives a poisoned TLP in the PCI Express* port. This bit is set regardless of the state the Parity Error Response Enable bit in the Bridge Control register.
14:14	RW1C	0x0	rse: Received System Error This bit is set by the root port when it receives a ERR_FATAL or ERR_NONFATAL message from PCI Express*. **Note:** This does not include the virtual ERR* messages that are internally generated from the root port when it detects an error on its own.
13:13	RW1C	0x0	rma: Received Master Abort Status This bit is set when the root port receives a Completion with 'Unsupported Request Completion' Status or when the root port master aborts a Type0 configuration packet that has a non-zero device number.
12:12	RW1C	0x0	rta: Received Target Abort Status This bit is set when the root port receives a Completion with 'Completer Abort' Status.
11:11	RW1C	0x0	sta: Signaled Target Abort This bit is set when the root port sends a completion packet with a 'Completer Abort' Status (including peer-to-peer completions that are forwarded from one port to another).
10:9	RO	0x0	devsel_timing: Not applicable to PCI Express*. Hardwired to 0.
8:8	RW1C	0x0	mdpe: Master Data Parity Error This bit is set by the root port on the secondary side (PCI Express* link) if the Parity Error Response Enable bit (PERRE) is set in Bridge Control register and either of the following two conditions occurs: The PCI Express* port receives a Completion from PCI Express* marked poisoned. The PCI Express* port poisons an outgoing packet with data. If the Parity Error Response Enable bit in Bridge Control Register is cleared, this bit is never set.
7:7	RO	0x0	fast_back_to_back_transactions_capable: Not applicable to PCI Express*. Hardwired to 0.
5:5	RO	0x0	pci66_mhz_capability: Not applicable to PCI Express*. Hardwired to 0.

7.2.17 mbas

Memory Base.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x20**				

Bit	Attr	Default	Description
15:4	RW	0xfff	memory_base_address: Corresponds to A[31:20] of the 32-bit memory windows base address of the PCI Express* port. See also the MLIM register description.

7.2.18 mlim

Memory Limit Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x22**				

Bit	Attr	Default	Description
15:4	RW	0x0	memory_limit_address: Corresponds to A[31:20] of the 32-bit memory window's limit address that corresponds to the upper limit of the range of memory accesses that will be passed by the PCI Express* bridge.The Memory Base and Memory Limit registers define a memory mapped I/O non-prefetchable address range (32-bit addresses) and the IIO directs accesses in this range to the PCI Express* port based on the following formula: MEMORY_BASE <= A[31:20] <= MEMORY_LIMIT The upper 12 bits of both the Memory Base and Memory Limit registers are read write and corresponds to the upper 12 address bits, A[31:20] of 32-bit addresses. Thus, the bottom of the defined memory address range will be aligned to a 1MB boundary and the top of the defined memory address range will be one less than a 1MB boundary. **Note:**Setting the memory limit less than memory base disables the 32-bit memory range altogether. **Note:** In general the memory base and limit registers won't be programmed by software without clearing the MSE bit first.

7.2.19 pbas

Prefetchable Memory Base Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x24**				

Bit	Attr	Default	Description
15:4	RW	0xfff	prefetchable_memory_base_address: Corresponds to A[31:20] of the prefetchable memory address range's base address of the PCI Express* port. See also the PLIMU register description.
3:0	RO	0x1	prefetchable_memory_base_address_capability: IIO sets this bit to 01h to indicate 64bit capability.

7.2.20 plim

Prefetchable Memory Limit Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x26**				

Bit	Attr	Default	Description
15:4	RW	0x0	prefetchable_memory_limit_address: Corresponds to A[31:20] of the prefetchable memory address range's limit address of the PCI Express* port. See also the PLIMU register description.
3:0	RO	0x1	prefetchable_memory_limit_address_capability: IIO sets this field to 01h to indicate 64bit capability.

7.2.21 pbasu

Prefetchable Memory Base Upper 32 bits

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x28**				

Bit	Attr	Default	Description
31:0	RW	0xffffffff	prefetchable_upper_32_bit_memory_base_address: Corresponds to A[63:32] of the prefetchable memory address range's base address of the PCI Express* port. See also the PLIMU register description.

7.2.22 plimu

Prefetchable Memory Limit Upper 32 bits.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x2c**				

Bit	Attr	Default	Description
31:0	RW	0x0	prefetchable_upper_32_bit_memory_limit_address: Corresponds to A[63:32] of the prefetchable memory address range's limit address of the PCI Express* port.The Prefetchable Memory Base and Memory Limit registers define a memory mapped I/O prefetchable address range (64-bit addresses) which is used by the PCI Express* bridge to determine when to forward memory transactions based on the following formula: PREFETCH_MEMORY_BASE_UPPER :: PREFETCH_MEMORY_BASE <= A[63:20] <= PREFETCH_MEMORY_LIMIT_UPPER :: PREFETCH_MEMORY_LIMIT The upper 12 bits of both the Prefetchable Memory Base and Memory Limit registers are read/write and corresponds to the upper 12 address bits, A[31:20] of 32-bit addresses. The bottom of the defined memory address range will be aligned to a 1MB boundary and the top of the defined memory address range will be one less than a 1MB boundary. The bottom 4 bits of both the Prefetchable Memory Base and Prefetchable Memory Limit registers are read-only, contain the same value, and encode whether or not the bridge supports 64-bit addresses. If these four bits have the value 0h, then the bridge supports only 32-bit addresses. If these four bits have the value 1h, then the bridge supports 64-bit addresses and the Prefetchable Base Upper 32 Bits and Prefetchable Limit Upper 32 Bits registers hold the rest of the 64-bit prefetchable base and limit addresses respectively. Setting the prefetchable memory limit less than prefetchable memory base disables the 64-bit prefetchable memory range altogether. **Note:** In general the memory base and limit registers won't be programmed by software without clearing the MSE bit first.

7.2.23 capptr

Capability Pointer

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x34**				

Bit	Attr	Default	Description
7:0	RO-V (Device 0 Function 0, Device 2 Function 0-3) RW-V (Device 3 Function 0) RO (Device 3 Function 1-3)	0x40 0x60 (Device 3 Function 0) 0x90 (Device 0 Function 0)	capability_pointer: Points to the first capability structure for the device which is the PCIe capability.

7.2.24 intl

Interrupt Line Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x3c**				

Bit	Attr	Default	Description
7:0	RW RO (Device 0 Function 0)	0x0	interrupt_line: N/A for these devices

7.2.25 intpin

Interrupt Pin Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x3d**				

Bit	Attr	Default	Description
7:0	RW-O	0x1	intp: N/A since these devices do not generate any interrupt on their own

7.2.26 bctrl

Bridge Control Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**1**	**Function:**	**0-1**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x3e**				

Bit	Attr	Default	Description
6:6	RW	0x0	sbr: 1: Setting this bit triggers a hot reset on the link for the corresponding PCI Express* port and the PCI Express* hierarchy domain subordinate to the port. This sends the LTSSM into the Training (or Link) Control Reset state, which necessarily implies a reset to the downstream device and all subordinate devices. The transaction layer corresponding to port will be emptied by virtue of the link going down when this bit is set. This means that in the outbound direction, all posted transactions are dropped and non-posted transactions are sent a UR response. In the inbound direction, completions for inbound NP requests are dropped when they arrive. Inbound posted writes are retired normally. **Note:** A secondary bus reset will not reset the virtual PCI-to-PCI bridge configuration registers of the targeted PCI Express* port. 0: No reset happens on the PCI Express* port.
4:4	RW	0x0	vga16b: This bit enables the virtual PCI-to-PCI bridge to provide 16-bit decoding of VGA I/O address precluding the decoding of alias addresses every 1KB. 0: execute 10-bit address decodes on VGA I/O accesses. 1: execute 16-bit address decodes on VGA I/O accesses. **Notes:** • This bit only has meaning if bit 3 of this register is also set to 1, enabling VGA I/O decoding and forwarding by the bridge. • Refer to PCI-PCI Bridge Specification Revision 1.2 for further details of this bit behavior.
3:3	RW	0x0	vgaen: Controls the routing of processor initiated transactions targeting VGA compatible I/O and memory address ranges. This bit must only be set for one p2p port in the entire system.
2:2	RW	0x0	isaen: Modifies the response by the root port to an I/O access issued by the core that target ISA I/O addresses. This applies only to I/O addresses that are enabled by the IOBASE and IOLIM registers. 1: The root port will not forward to PCI Express* any I/O transactions addressing the last 768 bytes in each 1KB block even if the addresses are within the range defined by the IOBASE and IOLIM registers. 0: All addresses defined by the IOBASE and IOLIM for core issued I/O transactions will be mapped to PCI Express*.
1:1	RW	0x0	serre: SERR Response Enable This bit controls forwarding of ERR_COR, ERR_NONFATAL and ERR_FATAL messages from the PCI Express* port to the primary side. 1: Enables forwarding of ERR_COR, ERR_NONFATAL and ERR_FATAL messages. 0: Disables forwarding of ERR_COR, ERR_NONFATAL and ERR_FATAL
0:0	RW	0x0	perre: Parity Error Response Enable This only effect this bit has is on the setting of bit 8 in the SECSTS register

7.2.27 scapid

Subsystem Capability Identity

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x40**				

Bit	Attr	Default	Description
7:0	RO RW-O (Device 0 Function 0)	0xd	capability_id: Assigned by PCI-SIG for subsystem capability ID

7.2.28 snxtptr

Subsystem ID Next Pointer

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x41**				

Bit	Attr	Default	Description
7:0	RO	0x60	next_ptr: This field is set to 60h for the next capability list MSI capability structure in the chain.

7.2.29 svid

Subsystem Vendor ID

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x2c**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x44**				

Bit	Attr	Default	Description
15:0	RW-O	0x8086	subsystem_vendor_id: Assigned by PCI-SIG for the subsystem vendor.

7.2.30 sdid

Subsystem Identity

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x2e**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x46**				

Bit	Attr	Default	Description
15:0	RW-O	0x0	subsystem_device_id: Assigned by the subsystem vendor to uniquely identify the subsystem.

7.2.31 dmircbar

DMI Root Complex Register Block Base Address.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x50**				

Bit	Attr	Default	Description
31:12	RW-LB	0x0	dmircbar: This field corresponds to bits 32 to 12 of the base address DMI Root Complex register space. The BIOS will program this register resulting in a base address for a 4KB block of contiguous memory address space. This register ensures that a naturally aligned 4KB space is allocated within the first 64GB of addressable memory space. System Software uses this base address to program the DMI Root Complex register set. All the Bits in this register are locked in Intel TXT mode. **Note:** This register is kept around on Device#0 even if that port is operating as PCIe port, to provide flexibility of using the VCs in PCIe mode as well.Nobody is asking for this capability right now but maintaining that flexibility.
0:0	RW-LB	0x0	dmircbaren: 0: DMIRCBAR is disabled and does not claim any memory 1: DMIRCBAR memory mapped accesses are claimed and decoded **Notes:** • Accesses to registers pointed to by the DMIRCBAR, by means of message channel or JTAG mini-port are not gated by this enable bit that is, accesses these registers are honored regardless of the setting of this bit. • The BIOS sets this bit only when it wishes to update the registers in the DMIRCBAR. It must clear this bit when it has finished changing values. This is required to ensure that the registers cannot be changed during an Intel TXT lock. This bit is protected by Intel TXT, but the registers in DMIRCBAR are not protected except by this bit.

7.2.32 msicapid

MSI Capability ID

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x60**				

Bit	Attr	Default	Description
7:0	RO	0x5	capability_id: Assigned by PCI-SIG for MSI root ports.

7.2.33 msinxtptr

MSI Next Pointer

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x61**				

Bit	Attr	Default	Description
7:0	RW-O	0x90	next_ptr: This field is set to 90h for the next capability list PCI Express* capability structure in the chain.

7.2.34 msimsgctl

MSI Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x62**				

Bit	Attr	Default	Description
8:8	RO	0x1	pvmc: This bit indicates that PCI Express* ports support MSI per-vector masking.
7:7	RO	0x0	b64ac: This field is hardwired to 0h since the message addresses are only 32-bit addresses (for example, FEEx_xxxxh).

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x62**				

Bit	Attr	Default	Description
6:4	RW	0x0	mme: Multiple Message Enable Applicable only to PCI Express* ports. Software writes to this field to indicate the number of allocated messages which is aligned to a power of two. When MSI is enabled, the software will allocate at least one message to the device. A value of 000 indicates 1 message. Any value greater than or equal to 001 indicates a message of 2. See MSIDR for discussion on how the interrupts are distributed amongst the various sources of interrupt based on the number of messages allocated by software for the PCI Express* ports.
3:1	RO	0x1	mmc: Multiple Message Capable The processor's Express* ports support two messages for all their internal events.
0:0	RW	0x0	msien: Software sets this bit to select INTx style interrupt or MSI interrupt for root port generated interrupts. 0: INTx interrupt mechanism is used for root port interrupts, provided the override bits in MISCCTRLSTS allow it 1: MSI interrupt mechanism is used for root port interrupts, provided the override bits in MISCCTRLSTS allow it **Note:** The bits 4:2 and bit 2 MISCCTRLSTS can disable both MSI and INTx interrupt from being generated on root port interrupt events.

7.2.35 msgadr

The MSI Address Register (MSIAR) contains the system specific address information to route MSI interrupts from the root ports and is broken into its constituent fields.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x64**				

Bit	Attr	Default	Description
31:2	RW	0x0	address_id: The definition of this field depends on whether interrupt remapping is enabled or disabled.

7.2.36 msgdat

MSI Data Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x68**				

Bit	Attr	Default	Description
15:0	RW	0x0	data: The definition of this field depends on whether interrupt remapping is enabled or disabled.

7.2.37 msimsk

MSI Mask Bit

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x6c**				

Bit	Attr	Default	Description
1:0	RW	0x0	mask_bits: Relevant only when MSI is enabled and used for interrupts generated by the root port. For each Mask bit that is set, the PCI Express* port is prohibited from sending the associated message. When only one message is allocated to the root port by software, only mask bit 0 is relevant and used by hardware.

7.2.38 msipending

MSI Pending Bit.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x70**				

Bit	Attr	Default	Description
1:0	RO-V	0x0	pending_bits: Relevant only when MSI is enabled and used for interrupts generated by the root port. When MSI is not enabled or used by the root port, this register always reads a value 0. For each Pending bit that is set, the PCI Express* port has a pending associated message. When only one message is allocated to the root port by software, only pending bit 0 is set cleared by hardware and pending bit 1 always reads 0. Hardware sets this bit whenever it has an interrupt pending to be sent. This bit remains set till either the interrupt is sent by hardware or the status bits associated with the interrupt condition are cleared by software.

7.2.39 pxpcapid

PCI Express Capability Identity

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x90**				

Bit	Attr	Default	Description
7:0	RO	0x10	capability_id: Provides the PCI Express* capability ID assigned by PCI-SIG.

7.2.40 pxpnxtptr

PCI Express* Next Pointer

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x91**				

Bit	Attr	Default	Description
7:0	RO	0xe0	next_ptr: This field is set to the PCI PM capability.

7.2.41 pxpcap

PCI Express Capabilities Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x92**				

Bit	Attr	Default	Description
13:9	RO	0x0	interrupt_message_number: Applies to root ports. This field indicates the interrupt message number that is generated for PM/HP/BW-change events. When there are more than one MSI interrupt Number allocated for the root port MSI interrupts, this register field is required to contain the offset between the base Message Data and the MSI Message that is generated when there are PM/HP/BW-change interrupts. IIO assigns the first vector for PM/HP/BW-change events and so this field is set to 0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x92**				

Bit	Attr	Default	Description
8:8	RW-O	0x0	slot_implemented: Applies only to the root ports. 1: indicates that the PCI Express* link associated with the port is connected to a slot. 0: indicates no slot is connected to this port. **Note:** This register bit is of type 'write once' and is set by the BIOS.
7:4	RO-V	0x4	device_port_type: This field identifies the type of device. It is set to 0x4 for all the Express* ports.
3:0	RW-O	0x2	capability_version: This field identifies the version of the PCI Express* capability structure, which is 2h as of now. This register field is left as RW-O to cover any unknowns with Gen3.

7.2.42 devcap

The PCI Express* Device Capabilities register identifies device specific information for the device.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x94**				

Bit	Attr	Default	Description
27:26	RO	0x0	captured_slot_power_limit_scale: Does not apply to root ports or integrated devices.
25:18	RO	0x0	captured_slot_power_limit_value: Does not apply to root ports or integrated devices.
15:15	RO	0x1	role_based_error_reporting: IIO is 1.1 compliant and so supports this feature
14:14	RO	0x0	power_indicator_present_on_device: Does not apply to root ports or integrated devices.
13:13	RO	0x0	attention_indicator_present: Does not apply to root ports or integrated devices.
12:12	RO	0x0	attention_button_present: Does not apply to root ports or integrated devices.
11:9	RO	0x0	endpoint_l1_acceptable_latency: N/A
8:6	RO	0x0	endpoint_l0s_acceptable_latency: N/A

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x94**				

Bit	Attr	Default	Description
5:5	RW-O	0x0 0x1 (Device 3 Function 0)	extended_tag_field_supported:
4:3	RO	0x0	phantom_functions_supported: IIO does not support phantom functions.
2:0	RO	0x1 0x0 (Device 0 Function 0)	max_payload_size_supported: Maximum payload is 128B on the DMI/PCIe port corresponding to Port 0.

7.2.43 devctrl

PCI Express Device Control

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0xf0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x98**				

Bit	Attr	Default	Description
14:12	RO	0x0	max_read_request_size: PCI Express*/DMI ports in Processor do not generate requests greater than 64B and this field is RO.
11:11	RO	0x0	enable_no_snoop: Not applicable to DMI or PCIe root ports since they never set the ‘No Snoop’ bit for transactions they originate (not forwarded from peer) to PCI Express*/DMI. This bit has no impact on forwarding of NoSnoop attribute on peer requests.
10:10	RO	0x0	auxiliary_power_management_enable: Not applicable to Processor
9:9	RO	0x0	phantom_functions_enable: Not applicable to IIO since it never uses phantom functions as a requester.
8:8	RW RO (Device 0 Function 0)	0x0	extended_tag_field_enable: N/A since IIO it never generates any requests on its own that uses tags 7:5. **Note:** On peer-to-peer writes, IIO forwards the tag field along without modification and tag fields 7:5 could be set and that is not impacted by this bit.
7:5	RW_LV RW (Device 0 Function 0)	0x0	max_payload_size: 000: 128B maximum payload size 001: 256B maximum payload size others: alias to 128B IIO can receive packets equal to the size set by this field. IIO generate read completions as large as the value set by this field. IIO generates memory writes of maximum 64B.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0xf0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x98**				

Bit	Attr	Default	Description
4:4	RO	0x0	enable_relaxed_ordering: Not applicable to root/DMI ports since they never set relaxed ordering bit as a requester (this does not include tx forwarded from peer devices). This bit has no impact on forwarding of relaxed ordering attribute on peer requests.
3:3	RW	0x0	unsupported_request_reporting_enable: This bit controls the reporting of unsupported requests that IIO itself detects on requests its receives from a PCI Express*/DMI port. 0: Reporting of unsupported requests is disabled 1: Reporting of unsupported requests is enabled.
2:2	RW	0x0	fatal_error_reporting_enable: Controls the reporting of fatal errors that IIO detects on the PCI Express*/DMI interface. 0: Reporting of Fatal error detected by device is disabled 1: Reporting of Fatal error detected by device is enabled Refer to PCI Express* Base Specification, Revision 2.0 for complete details of how this bit is used in conjunction with other bits to report errors. This bit is not used to control the reporting of other internal component uncorrectable fatal errors (at the port unit) in any way.
1:1	RW	0x0	non_fatal_error_reporting_enable: Controls the reporting of non-fatal errors that IIO detects on the PCI Express*/DMI interface. 0: Reporting of Non Fatal error detected by device is disabled 1: Reporting of Non Fatal error detected by device is enabled Refer to PCI Express* Base Specification, Revision 2.0 for complete details of how this bit is used in conjunction with other bits to report errors. This bit is not used to control the reporting of other internal component uncorrectable non-fatal errors (at the port unit) in any way.
0:0	RW	0x0	correctable_error_reporting_enable: Controls the reporting of correctable errors that IIO detects on the PCI Express*/DMI interface 0: Reporting of link Correctable error detected by the port is disabled 1: Reporting of link Correctable error detected by port is enabled Refer to PCI Express* Base Specification, Revision 2.0 for complete details of how this bit is used in conjunction with other bits to report errors. This bit is not used to control the reporting of other internal component correctable errors (at the port unit) in any way.

7.2.44 devsts

PCI Express* Device Status

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0xf2**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x9a**				

Bit	Attr	Default	Description
5:5	RO	0x0	transactions_pending: Does not apply to Root/DMI ports, that is, bit hardwired to 0 for these devices.
4:4	RO	0x0	aux_power_detected: Does not apply to the processor
3:3	RW1C	0x0	unsupported_request_detected: This bit indicates that the root port or DMI port detected an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register. 1: Unsupported Request detected at the device/port. These unsupported requests are NP requests inbound that the root port or DMI port received and it detected them as unsupported requests (for example, address decoding failures that the root port detected on a packet, receiving inbound lock reads, BME bit is clear and so forth). 0: No unsupported request detected by the root or DMI port **Note**: This bit is not set on peer-to-peer completions with UR status that are forwarded by the root port or DMI port to the PCIe*/DMI link.
2:2	RW1C	0x0	fatal_error_detected: This bit indicates that a fatal (uncorrectable) error is detected by the root or DMI port. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control register. 1: Fatal errors detected 0: No Fatal errors detected
1:1	RW1C	0x0	non_fatal_error_detected: This bit gets set if a non-fatal uncorrectable error is detected by the root or DMI port. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control register. 1: Non Fatal errors detected 0: No non-Fatal Errors detected
0:0	RW1C	0x0	correctable_error_detected: This bit gets set if a correctable error is detected by the root or DMI port. Errors are logged in this register regardless of whether error reporting is enabled or not in the PCI Express* Device Control register. 1: correctable errors detected 0: No correctable errors detected

7.2.45 lnkcap

PCI Express* Link Capabilities

The Link Capabilities register identifies the PCI Express* specific link capabilities. The link capabilities register needs some default values setup by the local host.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x9c**				

Bit	Attr	Default	Description
31:24	RW-O	0x0	port_number: This field indicates the PCI Express* port number for the link and is initialized by software/BIOS. IIO hardware does nothing with this bit.
22:22	RW-O	0x1	aspm_optionality_compliance:
21:21	RO-V	0x1	link_bandwidth_notification_capability: A value of 1b indicates support for the Link Bandwidth Notification status and interrupt mechanisms. **Note:** This bit will only be set if either "Report Speed Change" or "Report Configuration Change" bits are set in the DBG2STAT register bits 22 and 20 respectively.
20:20	RO	0x1	data_link_layer_link_active_reporting_capable: IIO supports reporting status of the data link layer so software knows when it can enumerate a device on the link or otherwise know the status of the link.
19:19	RO	0x1	surprise_down_error_reporting_capable: IIO supports reporting a surprise down error condition
18:18	RO	0x0	clock_power_management: Does not apply to processor
17:15	RW-O	0x2	l1_exit_latency: This field indicates the L1 exit latency for the given PCI-Express* port. It indicates the length of time this port requires to complete transition from L1 to L0. 000: Less than 1 µs 001: 1 µs to less than 2 µs 010: 2 µs to less than 4 µs 011: 4 µs to less than 8 µs 100: 8 µs to less than 16 µs 101: 16 µs to less than 32 µs 110: 32 µs to 64 µs 111: More than 64us This register is made writable once by the BIOS so that the value is settable based on experiments post-si.
14:12	RW-O	0x3	l0s_exit_latency: This field indicates the L0s exit latency (i.e L0s to L0) for the PCI-Express* port. 000: Less than 64 ns 001: 64 ns to less than 128 ns 010: 128 ns to less than 256 ns 011: 256 ns to less than 512 ns 100: 512 ns to less than 1 µs 101: 1 is to less than 2 µs 110: 2 is to 4 µs 111: More than 4 µs This register is made writable once by the BIOS so that the value is settable based on experiments post-si.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x9c**				

Bit	Attr	Default	Description
11:10	RW-O	0x3	active_state_link_pm_support: This field indicates the level of active state power management supported on the given PCI-Express* port. 00: Disabled 01: L0s Entry Supported 10: Reserved 11: L0s and L1 Supported
9:4	RW-O	0x4	maximum_link_width: This field indicates the maximum width of the given PCI Express* Link attached to the port. 000001: x1 000010: x2 000100: x4 001000: x8 010000: x16 Others: Reserved This is left as a RW-O register for the BIOS to update based on the platform usage of the links.
3:0	RW-O	0x3 0x1 (Device 0 Function 0)	maxlnkspd: This field indicates the maximum link speed of this Port. The encoding is the binary value of the bit location in the Supported Link Speeds Vector in LNKCAP2 that corresponds to the maximum link speed. The processor supports a maximum of 8Gbps, unless restricted by the Gen3OFF fuse. If Gen3OFF fuse is '1', this field defaults to 0010b 5Gbps If Gen3OFF fuse is '0' this field defaults to 0011b 8Gbps

7.2.46 lnkcon

PCI Express* Link Control

The PCI Express* Link Control register controls the PCI Express* Link specific parameters. The link control register needs some default values setup by the local host.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1b0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa0**				

Bit	Attr	Default	Description
11:11	RW	0x0	link_autonomous_bandwidth_interrupt_enable: For root ports, when set to 1b this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been set.For DMI mode on Dev#0, interrupt is not supported and hence this bit is not useful. Expectation is that The BIOS will set bit 27 in MISCCTRLSTS to notify the system of autonomous BW change event on that port.
10:10	RW	0x0	link_bandwidth_management_interrupt_enable: For root ports, when set to 1b this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been set.For DMI mode on Dev#0, interrupt is not supported and hence this bit is not useful. Expectation is that the BIOS will set bit 27 in MISCCTRLSTS to notify the system of autonomous BW change event on that port.
9:9	RW	0x0	hardware_autonomous_width_disable: When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width. **Note:** IIO does not by itself change width for any reason other than reliability. So this bit only disables such a width change as initiated by the device on the other end of the link.
8:8	RO	0x0	enable_clock_power_management:
7:7	RW	0x0	extended_synch: This bit when set forces the transmission of additional ordered sets when exiting L0s and when in recovery. See PCI Express* Base Specification, Revision 2.0 for details.
6:6	RW-V (Function 0) RW (Function 1-3)	0x0	common_clock_configuration: Software sets this bit to indicate that this component and the component at the opposite end of the Link are operating with a common clock source. A value of 0b indicates. that this component and the component at the opposite end of the Link are operating with separate reference clock sources. Default value of this bit is 0b. Components utilize this common clock configuration information to report the correct L0s and L1 Exit Latencies in the NFTS. The values used come from these registers depending on the value of this bit: 0: Use NFTS values from CLSPHYCTL3 1: Use NFTS values from CLSPHYCTL4

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1b0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa0**				

Bit	Attr	Default	Description
5:5	WO	0x0	retrain_link: A write of 1 to this bit initiates link retraining in the given PCI Express*/DMI port by directing the LTSSM to the recovery state if the current state is [L0, L0s, or L1]. If the current state is anything other than L0, L0s, L1 then a write to this bit does nothing. This bit always returns 0 when read.It is permitted to write 1b to this bit while simultaneously writing modified values to other fields in this register. If the LTSSM is not already in Recovery or Configuration, the resulting Link training must use the modified values. If the LTSSM is already in Recovery or Configuration, the modified values are not required to affect the Link training that's already in progress.
4:4	RW	0x0	link_disable: This field controls whether the link associated with the PCI Express*/DMI port is enabled or disabled. When this bit is a 1, a previously configured link would return to the ’disabled’ state as defined in the PCI Express* Base Specification, Revision 2.0. When this bit is clear, an LTSSM in the ’disabled’ state goes back to the detect state. 0: Enables the link associated with the PCI Express* port 1: Disables the link associated with the PCI Express* port
3:3	RO	0x0	read_completion_boundary: Set to zero to indicate IIO could return read completions at 64B boundaries
1:0	RW-V (Function 0) RW (Function 1-3)	0x0	active_state_link_pm_control: When 01b or 11b, L0s on transmitter is enabled, otherwise it is disabled. 10 and 11 enables L1 ASPM.

7.2.47 lnksts

PCI Express* Link Status

The PCI Express* Link Status register provides information on the status of the PCI Express* Link such as negotiated width, training, and so forth. The link status register needs some default values setup by the local host.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1b2**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa2**				

Bit	Attr	Default	Description
15:15	RW1C	0x0	link_autonomous_bandwidth_status: This bit is set to 1b by hardware to indicate that hardware has autonomously changed link speed or width, without the port transitioning through DL_Down status, for reasons other than to attempt to correct unreliable link operation. IIO does not, on its own, change speed or width autonomously for non-reliability reasons. IIO only sets this bit when it receives a width or speed change indication from downstream component that is not for link reliability reasons.
14:14	RW1C	0x0	link_bandwidth_management_status: This bit is set to 1b by hardware to indicate that either of the following has occurred without the port transitioning through DL_Down status: a) A link retraining initiated by a write of 1b to the Retrain Link bit has completed b) Hardware has autonomously changed link speed or width to attempt to correct unreliable link operation **Note:** IIO also sets this bit when it receives a width or speed change indication from downstream component that is for link reliability reasons.
13:13	RO-V	0x0	data_link_layer_link_active: Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.When this bit is 0b, the transaction layer associated with the link will abort all transactions that would otherwise be routed to that link.
12:12	RW-O	0x1	slot_clock_configuration: This bit indicates whether the processor receives clock from the same xtal that also provides clock to the device on the other end of the link. 1: indicates that same xtal provides clocks to the processor and the slot or device on other end of the link 0: indicates that different xtals provide clocks to the processor and the slot or device on other end of the link In general, this field is expected to be set to 1b by the BIOS based on board clock routing. This bit has to be set to 1b on DMI mode operation on Device#0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1b2**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa2**				

Bit	Attr	Default	Description
11:11	RO-V	0x0	link_training: This field indicates the status of an ongoing link training session in the PCI Express* port 0: LTSSM has exited the recovery/configuration state. 1: LTSSM is in recovery/configuration state or the Retrain Link was set but training has not yet begun. The IIO hardware clears this bit once LTSSM has exited the recovery/configuration state. Refer to PCI Express* Base Specification, Revision 2.0 for details of which states within the LTSSM would set this bit and which states would clear this bit.
9:4	RO-V	0x0	negotiated_link_width: This field indicates the negotiated width of the given PCI Express* link after training is completed. Only x1, x2, x4, x8 and x16 link width negotiations are possible in the processor for Device#1-2 and only x1, x2 and x4 on Device#0. A value of 0x01 in this field corresponds to a link width of x1, 0x02 indicates a link width of x2 and so on, with a value of 0x10 for a link width of x16.The value in this field is reserved and could show any value when the link is not up. Software determines if the link is up or not by reading bit 13 of this register.
3:0	RO-V	0x1	current_link_speed:

7.2.48 sltcap

PCI Express* Slot Capabilities

The Slot Capabilities register identifies the PCI Express* specific slot capabilities.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa4**				

Bit	Attr	Default	Description
31:19	RW-O	0x0	physical_slot_number: This field indicates the physical slot number of the slot connected to the PCI Express* port and is initialized by the BIOS.
18:18	RO	0x0	command_complete_not_capable: The processor is capable of command complete interrupt.
17:17	RW-O	0x0	electromechanical_interlock_present: This bit when set indicates that an Electromechanical Interlock is implemented on the chassis for this slot and that lock is controlled by bit 11 in Slot Control register. This field is initialized by the BIOS based on the System Architecture.BIOS. **Note:** This capability is not set if the Electromechanical Interlock control is connected to main slot power control. This is expected to be used only for Express* Module hotpluggable slots.
16:15	RW-O	0x0	slot_power_limit_scale:
14:7	RW-O	0x0	slot_power_limit_value:

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa4**				

Bit	Attr	Default	Description
6:6	RW-O	0x0	hot_plug_capable: This field defines hot-plug support capabilities for the PCI Express* port. 0: indicates that this slot is not capable of supporting Hot-plug operations. 1: indicates that this slot is capable of supporting Hot-plug operations This bit is programed by the BIOS based on the system design. This bit must be programmed by the BIOS to be consistent with the VPP enable bit for the port.
5:5	RW-O	0x0	hot_plug_surprise: This field indicates that a device in this slot may be removed from the system without prior notification. This field is initialized by the BIOS. 0: indicates that hot-plug surprise is not supported 1: indicates that hot-plug surprise is supported Generally this bit is not expected to be set because the only know usage case for this is the Express*Card FF. But that is not really expected usage in the processor context. But this bit is present regardless to allow a usage if it arises. This bit is used by IIO hardware to determine if a transition from DL_active to DL_Inactive is to be treated as a surprise down error or not. If a port is associated with a hotpluggable slot and the hot-plug surprise bit is set, then any transition to DLInactive is not considered an error. Refer to PCI Express* Base Specification, Revision 2.0 for further details.
4:4	RW-O	0x0	power_indicator_present: This bit indicates that a Power Indicator is implemented for this slot and is electrically controlled by the chassis. 0: indicates that a Power Indicator that is electrically controlled by the chassis is not present 1: indicates that Power Indicator that is electrically controlled by the chassis is present The BIOS programs this field with a 1 for CEMExpress* Module FFs, if the slot is hot-plug capable.
3:3	RW-O	0x0	attention_indicator_present: This bit indicates that an Attention Indicator is implemented for this slot and is electrically controlled by the chassis 0: indicates that an Attention Indicator that is electrically controlled by the chassis is not present 1: indicates that an Attention Indicator that is electrically controlled by the chassis is present The BIOS programs this field with a 1 for CEM Express Module FFs, if the slot is hot-plug capable.
2:2	RW-O	0x0	mrl_sensor_present: This bit indicates that an MRL Sensor is implemented on the chassis for this slot. 0: indicates that an MRL Sensor is not present 1: indicates that an MRL Sensor is present The BIOS programs this field with a 0 for Express Module FF always. If CEM slot is hot-plug capable, the BIOS programs this field with either 0 or 1 depending on system design.
1:1	RW-O	0x0	power_controller_present: This bit indicates that a software controllable power controller is implemented on the chassis for this slot. 0: indicates that a software controllable power controller is not present 1: indicates that a software controllable power controller is present The BIOS programs this field with a 1 for CEM Express Module FFs, if the slot is hot-plug capable.

Type: CFG Bus: 0 Bus: 0 Bus: 0 Offset: 0xa4		PortID: N/A Device: 0 Device: 2 Device: 3	Function: 0 (PCIe Mode) Function: 0-3 Function: 0-3
Bit	**Attr**	**Default**	**Description**
0:0	RW-O	0x0	attention_button_present: This bit indicates that the Attention Button event signal is routed from slot or on-board in the chassis to the IIO's hot-plug controller. 0: indicates that an Attention Button signal is routed to IIO 1: indicates that an Attention Button is not routed to IIO The BIOS programs this field with a 1 for CEM Express Module FFs, if the slot is hot-plug capable.

7.2.49 sltcon

PCI Express* Slot Control

Any write to this register will set the Command Completed bit in the SLTSTS register, ONLY if the VPP enable bit for the port is set. If the port's VPP enable bit is set that is, hot-plug for that slot is enabled, then the required actions on VPP are completed before the Command Completed bit is set in the SLTSTS register. If the VPP enable bit for the port is clear, then the write simply updates this register see individual bit definitions for details but the Command Completed bit in the SLTSTS register is not set.

Type: CFG Bus: 0 Bus: 0 Bus: 0 Offset: 0xa8		PortID: N/A Device: 0 Device: 2 Device: 3	Function: 0 (PCIe Mode) Function: 0-3 Function: 0-3
Bit	**Attr**	**Default**	**Description**
12:12	RWS	0x0	data_link_layer_state_changed_enable: When set to 1, this field enables software notification when Data Link Layer Link Active bit in the LNKSTS register changes state
11:11	RW	0x0	electromechanical_interlock_control: When software writes either a 1 to this bit, IIO pulses the EMIL pin per lt;Bluegt;PCI Express* ServerWorkstation Module Electromechanical Specification Rev 1.0. Write of 0 has no effect. This bit always returns a 0 when read. If electromechanical lock is not implemented, then either a write of 1 or 0 to this register has no effect.
10:10	RWS	0x1	power_controller_control: If a power controller is implemented, when writes to this field will set the power state of the slot per the defined encodings. Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 0: Power On 1: Power Off **Note**: If the link experiences an unexpected DL_Down condition that is not the result of a hot-plug removal, the processor follows the PCI Express* specification for logging Surprise Link Down. Software is required to set SLTCON[10] to 0 (Power On) in all devices that do not connect to a slot that supports Hot-Plug to enable logging of this error in that device. For devices connected to slots supporting Hot-Plug operations, SLTCON[10] usage to control PWREN# assertion is as described elsewhere.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa8**				

Bit	Attr	Default	Description
9:8	RW	0x3	power_indicator_control: If a Power Indicator is implemented, writes to this field will set the Power Indicator to the written state. Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 00: Reserved. 01: On 10: Blink (IIO drives 1 Hz square wave for Chassis mounted LEDs) 11: Off IIO does not generated the Power_Indicator_On/Off/Blink messages on PCI Express* when this field is written to by software.
7:6	RW	0x3	attention_indicator_control: If an Attention Indicator is implemented, writes to this field will set the Attention Indicator to the written state. Reads of this field reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 00: Reserved. 01: On 10: Blink (Processor drives 1 Hz square wave) 11: Off IIO does not generated the Attention_Indicator_On/Off/Blink messages on PCI Express* when this field is written to by software.
5:5	RW	0x0	hot_plug_interrupt_enable: When set to 1b, this bit enables generation of Hot-Plug interrupt MSI or INTx interrupt depending on the setting of the MSI enable bit in MSICTRL on enabled Hot-Plug events, provided ACPI mode for hot-plug is disabled. 0: disables interrupt generation on Hot-plug events 1: enables interrupt generation on Hot-plug events
4:4	RW	0x0	command_completed_interrupt_enable: This field enables software notification Interrupt - MSIINTx or WAKE when a command is completed by the Hot-plug controller connected to the PCI Express* port 0: disables hot-plug interrupts on a command completion by a hot-plug Controller 1: Enables hot-plug interrupts on a command completion by a hot-plug Controller
3:3	RW	0x0	presence_detect_changed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a presence detect changed event. 0: disables generation of hot-plug interrupts or wake messages when a presence detect changed event happens. 1- Enables generation of hot-plug interrupts or wake messages when a presence detect changed event happens.
2:2	RW	0x0	mrl_sensor_changed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a MRL Sensor changed event. 0: disables generation of hot-plug interrupts or wake messages when an MRL Sensor changed event happens. 1: Enables generation of hot-plug interrupts or wake messages when an MRL Sensor changed event happens.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xa8**				

Bit	Attr	Default	Description
1:1	RW	0x0	power_fault_detected_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a power fault event. 0: disables generation of hot-plug interrupts or wake messages when a power fault event happens. 1: Enables generation of hot-plug interrupts or wake messages when a power fault event happens.
0:0	RW	0x0	attention_button_pressed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of an attention button pressed event. 0: disables generation of hot-plug interrupts or wake messages when the attention button is pressed. 1: Enables generation of hot-plug interrupts or wake messages when the attention button is pressed.

7.2.50 sltsts

PCI Express* Slot Status

The PCI Express Slot Status register defines important status information for operations such as hot-plug and Power Management.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xaa**				

Bit	Attr	Default	Description
8:8	RW1C	0x0	data_link_layer_state_changed: This bit is set (if it is not already set) when the state of the Data Link Layer Link Active bit in the Link Status register changes. Software must read Data Link Layer Active field to determine the link state before initiating configuration cycles to the hot plugged device.
7:7	RO-V	0x0	electromechanical_latch_status: When read this register returns the current state of the Electromechanical Interlock (the EMILS pin) which has the defined encodings as: 0: Electromechanical Interlock Disengaged 1: Electromechanical Interlock Engaged

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xaa**				

Bit	Attr	Default	Description
6:6	RO-V	0x0	presence_detect_state: For ports with slots (where the Slot Implemented bit of the PCI Express* Capabilities Registers is 1b), this field is the logical OR of the Presence Detect status determined by means of an in-band mechanism and sideband Present Detect pins. Refer to how PCI Express* Base Specification, Revision 2.0 for how the inband presence detect mechanism works (certain states in the LTSSM constitute 'card present' and others don't). 0: Card/Module slot empty 1: Card/module Present in slot (powered or unpowered) For ports with no slots, IIO hardwires this bit to 1b. **Note**: The OS could get confused when it sees an empty PCI Express* root port that is, 'no slots + no presence', since this is now disallowed in the Specification. So the BIOS must hide all unused root ports devices in IIO config space, by means of the DEVHIDE register.
5:5	RO-V	0x0	mrl_sensor_state: This bit reports the status of an MRL sensor if it is implemented. 0: MRL Closed 1: MRL Open
4:4	RW1C	0x0	command_completed: This bit is set by IIO when the hot-plug command has completed and the hot-plug controller is ready to accept a subsequent command. It is subsequently cleared by software after the field has been read and processed. This bit provides no guarantee that the action corresponding to the command is complete.Any write to SLTCON (regardless of the port is capable or enabled for hot-plug) is considered a 'hot-plug' command. If the port is not hot-plug capable or hot-plug enabled, then the hot-plug command does not trigger any action on the VPP port but the command is still completed by means of this bit.
3:3	RW1C	0x0	presence_detect_changed: This bit is set by IIO when the value reported in bit 6 is changes. It is subsequently cleared by software after the field has been read and processed.
2:2	RW1C	0x0	mrl_sensor_changed: This bit is set if the value reported in bit 5 changes. It is subsequently cleared by software after the field has been read and processed.
1:1	RW1C	0x0	power_fault_detected: This bit is set by IIO when a power fault event is detected by the power controller (which is reported by means of the VPP bit stream). It is subsequently cleared by software after the field has been read and processed.
0:0	RW1C	0x0	attention_button_pressed: This bit is set by IIO when the attention button is pressed. It is subsequently cleared by software after the field has been read and processed. IIO silently discards the AttentionButtonPressed message if received from PCI Express* link without updating this bit.

7.2.51 rootcon

PCI Express* Root Control

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xac**				

Bit	Attr	Default	Description
4:4	RW	0x0	crsswvisen: CRS software visibility Enable This bit, when set, enables the Root Port to return Configuration Request Retry Status (CRS) Completion Status to software.
3:3	RW RW-L (Device 3 Function 0 only	0x0	pmeinten: This field controls the generation of MSI interruptsINTx interrupts for PME messages. 1: Enables interrupt generation upon receipt of a PME message 0: Disables interrupt generation for PME messages
2:2	RW	0x0	SEFEEN: System Error on Fatal Error Enable This field enables notifying the internal IIO core error logic of occurrence of an uncorrectable fatal error at the port or below its hierarchy. The internal core error logic of IIO then decides if/how to escalate the error further (pins/message and so forth). 1: indicates that an internal IIO core error logic notification should be generated if a fatal error (ERR_FATAL) is reported by any of the devices in the hierarchy associated with and including this port. 0: No internal IIO core error logic notification should be generated on a fatal error (ERR_FATAL) reported by any of the devices in the hierarchy associated with and including this port. **Note:** The generation of system notification on a PCI Express* fatal error is orthogonal to generation of an MSI/INTx interrupt for the same error. Both a system error and MSI/INTx can be generated on a fatal error or software can chose one of the two. Refer to PCI Express* Base Specification, Revision 2.0 for details of how this bit is used in conjunction with other error control bits to generate core logic notification of error events in a PCI Express* port. **Note:** Since this register is defined only in PCIe mode for Device#0, this bit will read a 0 in DMI mode. So, to enable core error logic notification on DMI mode fatal errors, the BIOS must set bit 35 of MISCCTRLSTS to a 1 (to override this bit) on Device#0 in DMI mode.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xac**				

Bit	Attr	Default	Description
1:1	RW	0x0	SENFEEN: System Error on Non-Fatal Error Enable This field enables notifying the internal IIO core error logic of occurrence of an uncorrectable non-fatal error at the port or below its hierarchy. The internal IIO core error logic then decides if/how to escalate the error further (pins/message and so forth). 1: indicates that a internal IIO core error logic notification should be generated if a non-fatal error (ERR_NONFATAL) is reported by any of the devices in the hierarchy associated with and including this port. 0: No internal core error logic notification should be generated on a non-fatal error (ERR_NONFATAL) reported by any of the devices in the hierarchy associated with and including this port. **Note:** The generation of system notification on a PCI Express* non-fatal error is orthogonal to generation of an MSI/INTx interrupt for the same error. Both a system error and MSI/INTx can be generated on a non-fatal error or software can chose one of the two. Refer to PCI Express* Base Specification, Revision 2.0 for details of how this bit is used in conjunction with other error control bits to generate core logic notification of error events in a PCI Express* port. **Note:** This register is defined only in PCIe mode for Device#0, this bit will read a 0 in DMI mode. So, to enable core error logic notification on DMI mode non-fatal errors, the BIOS must set bit 34 of MISCCTRLSTS to a 1 (to override this bit) on Device#0 in DMI mode.
0:0	RW	0x0	SECEEN: System Error on Correctable Error Enable This field controls notifying the internal IIO core error logic of the occurrence of a correctable error in the device or below its hierarchy. The internal core error logic of IIO then decides if/how to escalate the error further (pins/message and so forth). 1: indicates that an internal core error logic notification should be generated if a correctable error (ERR_COR) is reported by any of the devices in the hierarchy associated with and including this port. 0: No internal core error logic notification should be generated on a correctable error (ERR_COR) reported by any of the devices in the hierarchy associated with and including this port. **Note:** This generation of system notification on a PCI Express* correctable error is orthogonal to generation of an MSI/INTx interrupt for the same error. Both a system error and MSI/INTx can be generated on a correctable error or software can chose one of the two. Refer to PCI Express* Base Specification, Revision 2.0 for details of how this bit is used in conjunction with other error control bits to generate core logic notification of error events in a PCI Express* port. **Note:** This register is defined only in PCIe mode for Device#0, this bit will read a 0 in DMI mode. So, to enable core error logic notification on DMI mode correctable errors, the BIOS must set bit 33 of MISCCTRLSTS to a 1 (to override this bit) on Device#0 in DMI mode.

7.2.52 rootcap

PCI Express* Root Capabilities

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xae**				

Bit	Attr	Default	Description
0:0	RO RW-O (Device 0 Function 0)	0x1 0x0 (Device 0 Function 0, DMI2 mode)	crs_software_visibility: This bit, when set, indicates that the Root Port is capable of returning Configuration Request Retry Status (CRS) Completion Status to software. The processor supports this capability.

7.2.53 rootsts

PCI Express* Root Status

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xb0**				

Bit	Attr	Default	Description
17:17	RO-V	0x0	pme_pending: This field indicates that another PME is pending when the PME Status bit is set. When the PME Status bit is cleared by software; the pending PME is delivered by hardware by setting the PME Status bit again and updating the Requestor ID appropriately. The PME pending bit is cleared by hardware if no more PMEs are pending.
16:16	RW1C	0x0	pme_status: This field indicates a PM_PME message (either from the link or internally from within that root port) was received at the port. 1: PME was asserted by a requester as indicated by the PME Requester ID field This bit is cleared by software by writing a '1'. **Note:** The root port itself could be the source of a PME event when a hot-plug event is observed when the port is in D3hot state.
15:0	RO-V	0x0	pme_requester_id: This field indicates the PCI requester ID of the last PME requestor. If the root port itself was the source of the (virtual) PME message, then a RequesterID of CPUBUSNO0:DevNo:FunctionNo is logged in this field.

7.2.54 devcap2

PCI Express* Device Capabilities 2 Register

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xb4**				

Bit	Attr	Default	Description
13:12	RW-O	0x1	tph_completer_supported: Indicates the support for TLP Processing Hints. Processor does not support the extended TPH header. 00: TPH and Extended TPH Completer not supported. 01: TPH Completer supported; Extended TPH Completer not supported. 10: Reserved. 11: Both TPH and Extended TPH Completer supported.
9:9	RO	0x1	atomic128bcascompsup:
8:8	RO	0x1	atomic64bcompsup:
7:7	RO	0x1	atomic32bcompsup:
6:6	RO	0x0	atomicroutsup:
5:5	RW-O	0x1	ari_en: Alternative RID InterpretationCapable This bit is set to 1b indicating Root Port supports this capability.
4:4	RO	0x1	cmpltodissup: Completion Timeout Disable Supported IIO supports disabling completion timeout
3:0	RO	0xe	cmpltovalsup: Completion Timeout Values Supported This field indicates device support for the optional Completion Timeout programmability mechanism. This mechanism allows system software to modify the Completion Timeout range. Bits are one-hot encoded and set according to the table below to show timeout value ranges supported. A device that supports the optional capability of Completion Timeout Programmability must set at least two bits.Four time values ranges are defined: Range A: 50 µs to 10 ms Range B: 10 ms to 250 ms Range C: 250 ms to 4 s Range D: 4 sec. to 64 sec. Bits are set according to table below to show timeout value ranges supported. 0000b: Completions Timeout programming not supported – values is fixed by implementation in the range 50 µs to 50 ms. 0001b: Range A 0010b: Range B 0011b: Range A and amp; B 0110b: Range B and amp; C 0111b: Range A, B, and amp; C 1110b: Range B, C, D 1111b: Range A, B, C, and amp; D All other values are reserved. IIO supports timeout values up to 10 ms–64 seconds

7.2.55 devctrl2

PCI Express* Device Control Register 2

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0xf8**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xb8**				

Bit	Attr	Default	Description
7:7	RO	0x0	atomicegressblock:
6:6	RO	0x0	atomicreqen:
5:5	RW-L	0x0	ari: Alternative RID InterpretationEnable Applies only to root ports. When set to 1b, ARI is enabled for the Root Port. For Device#0 in DMI mode, this bit is ignored
4:4	RW-V (Device 2 and 3 Function 0) RW (Device 0 Function0, Device 2 and 3 Function 1-3)	0x0 0x1 (Device 0 Function 0)	compltodis: Completion Timeout Disable When set to 1b, this bit disables the Completion Timeout mechanism for all NP tx that IIO issues on the PCIe/DMI link. When 0b, completion timeout is enabled. Software can change this field while there is active traffic in the root/ DMI port.
3:0	RW-V (Device 2 and 3 Function 0) RW (Device 0 Function0, Device 2 and 3 Function 1-3)	0x0	compltoval: Completion Timeout Value on NP Tx that IIO issues on PCIe/DMI In Devices that support Completion Timeout programmability, this field allows system software to modify the Completion Timeout range. The following encodings and corresponding timeout ranges are defined: 0000b = 10–50 ms 0001b = Reserved (IIO aliases to 0000b) 0010b = Reserved (IIO aliases to 0000b) 0101b = 16–55 ms 0110b = 65–210 ms 1001b = 260–900 ms 1010b = 1–3.5 seconds 1101b = 4–13 seconds 1110b = 17–64 seconds When software selects 17–64 seconds range, CTOCTRL further controls the timeout value within that range. For all other ranges selected by OS, the timeout value within that range is fixed in IIO hardware. Software can change this field while there is active traffic in the root port. This value will also be used to control PME_TO_ACK Timeout. That is this field sets the timeout value for receiving a PME_TO_ACK message after a PME_TURN_OFF message has been transmitted. The PME_TO_ACK Timeout has meaning only if bit 6 of MISCCTRLSTS register is set to a 1b.

7.2.56 lnkcap2

PCI Express* Link Capabilities 2.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xbc**				

Bit	Attr	Default	Description
7:1	RW-O	0x7 0x3 (Device 0 Function 0)	lnkspdvec: Supported Link Speeds Vector - This field indicates the supported Link speeds of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit definitions are: Bit 1 2.5 GTs set in processor Bit 2 5.0 GTs set in processor Bit 3 8.0 GTs set in processor if Gen3OFF fuse is not blown Bits 7:4 reserved processor supports all speeds, unless Gen3OFF fuse is set, then only Gen1 and Gen2 are supported.

7.2.57 lnkcon2

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1c0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xc0**				

Bit	Attr	Default	Description
15:12 12:12 (Device 0 Function 0)	RWS	0x0	compliance_de_emphasis: For 8GT/s Data Rate: This bit sets the Transmitter Preset level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. The Encodings are defined as follows: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot Others: reserved For 5 GT/s Data Rate: This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 0001b: -3.5 dB 0000b: -6 dB For 2.5 GT/s Data Rate: The setting of this field has no effect. Components that support only 2.5 GT/s speed are permitted to hardwire this field to 0h. **Note:** This bit is intended for debug, compliance testing purposes. System firmware and software is allowed to modify this bit only during debug or compliance testing.
11:11	RWS	0x0	compliance_sos: When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns.
10:10	RWS	0x0	enter_modified_compliance: When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate.
9:7	RWS_V	0x0	transmit_margin: This field controls the value of the nondeemphasized voltage level at the Transmitter pins.
6:6	RW-O	0x0	selectable_de_emphasis: When the Link is operating at 5.0 GT/s speed, this bit selects the level of de-emphasis for an Upstream component.Encodings: 1b -3.5 dB 0b -6 dB When the Link is operating at 2.5 GT/s speed, the setting of this bit has no effect.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1c0**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xc0**				

Bit	Attr	Default	Description
5:5	RWS	0x0	hardware_autonomous_speed_disable: When Set, this bit disables hardware from changing the Link speed for device specific reasons other than attempting to correct unreliable Link operation by reducing Link speed.
4:4	RWS_V	0x0	enter_compliance: Software is permitted to force a link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a link and then initiating a hot reset on the link.
3:0	RWS_V	0x3 0x2 (Device 0 Function 0)	target_link_speed: This field sets an upper limit on link operational speed by restricting the values advertised by the upstream component in its training sequences. Defined encodings are: 0001b 2.5Gb/s Target Link Speed 0010b 5Gb/s Target Link Speed 0011b 8Gb/s Target Link Speed (Reserved for Device 0 Function 0) All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, IIO will default to Gen1 speed. This field is also used to set the target compliance mode speed when software is using the Enter Compliance bit to force a link into compliance mode.

7.2.58 lnksts2

PCI Express* Link Status Register 2.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x1c2**				
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xc2**				

Bit	Attr	Default	Description
5:5	RW1CS	0x0	lnkeqreq: This bit is Set by hardware to request Link equalization process to be performed on the link. Reserved for Device 0 Function 0.
4:4	RO-V	0x0	eqph3_succ: When set to 1b, this indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed. Reserved for Device 0 Function 0.
3:3	RO-V	0x0	eqph2_succ: When set to 1b, this indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed. Reserved for Device 0 Function 0.
2:2	RO-V	0x0	eqph1_succ: When set to 1b, this indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed. Reserved for Device 0 Function 0.
1:1	RO-V	0x0	eqcmp: When set to 1b, this indicates that the Transmitter Equalization procedure has completed. Reserved for Device 0 Function 0.
0:0	RO-V	0x0	current_de_emphasis_level: When operating at Gen2 speed, this reports the current de-emphasis level. This field is Unused for Gen1 speeds 1b: -3.5 dB 0b: -6 dB

7.2.59 pmcap

Power Management Capabilities

The PM Capabilities Register defines the capability ID, next pointer and other power management related support. The following PM registers/capabilities are added for software compliance.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xe0**				

Bit	Attr	Default	Description
31:27	RO-V	0x19	pme_support: For DMI it should be 0, 0x19 for the PCIE ports. Bits 31, 30 and 27 must be set to q1q for PCI-PCI bridge structures representing ports on root complexes.
26:26	RO	0x0	d2_support: IOxAPIC does not support power management state D2.
25:25	RO	0x0	d1_support: IOxAPIC does not support power management state D1.
24:22	RO	0x0	aux_current:
21:21	RO	0x0	device_specific_initialization:
19:19	RO	0x0	pme_clock: This field is hardwired to 0h as it does not apply to PCI Express*.
18:16	RO	0x3	version: This field is set to 3h PM 1.2 compliant as version number. Bit is RW-O to make the version 2h incase legacy OS'es have any issues.
15:8	RO	0x0	next_capability_pointer: This is the last capability in the chain and hence set to 0.
7:0	RO	0x1	capability_id: Provides the PM capability ID assigned by PCI-SIG.

7.2.60 pmcsr

Power Management Control and Status Register

This register provides status and control information for PM events in the PCI Express* port of the IIO.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xe4**				

Bit	Attr	Default	Description
31:24	RO	0x0	data: Not relevant for IOxAPIC
23:23	RO	0x0	bus_power_clock_control_enable: Not relevant for IOxAPIC

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0xe4**				

Bit	Attr	Default	Description
22:22	RO	0x0	b2_b3_support: Not relevant for IOxAPIC
15:15	RW1CS	0x0	pme_status: Not relevant for IOxAPIC
14:13	RO	0x0	data_scale: Not relevant for IOxAPIC
12:9	RO	0x0	data_select: Not relevant for IOxAPIC
8:8	RWS RWS_L (Device 3 Function 0)	0x0	pme_enable: Not relevant for IOxAPIC
3:3	RW-O	0x1	no_soft_reset: Indicates IOxAPIC does not reset its registers when transitioning from D3hot to D0.
1:0	RW RW-L (Device 0 Function 0)	0x0	power_state: This 2-bit field is used to determine the current power state of the function and to set a new power state as well. 00: D0 01: D1 (not supported by IIO) 10: D2 (not supported by IIO) 11: D3hot If Software tries to write 01 or 10 to this field, the power state does not change from the existing power state which is either D0 or D3hot and nor do these bits1:0 change value. When in D3hot state, IOxAPIC will: a) respond to only Type 0 configuration transactions targeted at the device's configuration space, when in D3hot state b) will not respond to memory that is, D3hot state is equivalent to MSE, accesses to MBAR region **Note:** ABAR region access still go through in D3hot state, if it enabled c) will not generate any MSI writes

7.2.61 xpreut_hdr_ext

REUT PCIe* Header Extended

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x100**				

Bit	Attr	Default	Description
31:20	RO RO-V (Device 0 Function 0)	0x110	pcienextptr: Next Capability Pointer This field contains the offset to the next PCI capability structure or 00h if no other items exist in the linked list of capabilities. In DMI Mode, it points to the Vendor Specific Error Capability. In PCIe Mode, it points to the ACS Capability.
19:16	RO	0x1	pciecapversion: Capability Version: This field is a PCI-SIG defined version number that indicates the nature and format of the extended capability. This indicates the version of the REUT Capability.
15:0	RO	0xb	pciecapid: PCIe Extended CapID: This field has the value 0Bh to identify the CAP_ID assigned by the PCI SIG indicating a vendor specific capability.

7.2.62 xpreut_hdr_cap

REUT PCIe Header Capability.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x104**				

Bit	Attr	Default	Description
31:20	RO	0xc	vseclength: VSEC Length This field defines the length of the REUT ‘capability body’. The size of the leaf body is 12 bytes including the _EXT, _CAP and _LEF registers
19:16	RO	0x0	vsecidrev: REUT VSECID Rev This field is defined as the version number that indicates the nature and format of the VSEC structure. Software must quality the Vendor ID before interpreting this field.
15:0	RO	0x2	vsecid: REUT Engine VSECID This field is a Intel-defined ID number that indicates the nature and format of the VSEC structure. Software must qualify the Vendor ID before interpreting this field. **Notes:** • A value of ‘00h’ is reserved • A value of ‘01h’ is the ID Council defined for REUT engines. • A value of ‘02h’ is specified for the REUT ‘leaf’ capability structure which resides in each link which in supported by a REUT engine.

7.2.63 xpreut_hdr_lef

REUT Header Leaf Capability.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x108**				

Bit	Attr	Default	Description

15:8	RO-V	0x38
		0x30 (Device 0 Function 0)
			leafreutdevnum:
			This field identifies the PCI Device/Function # where the REUT engine associated with this link resides.
			Device6 = 00110b and function0 = 000b = 30h

7:0	RO-V	0x7	leafreutengid: This field identifies the REUT engine associated with the link (same as the REUT ID).

7.2.64 acscaphdr

Access Control Services Extended Capability Header.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x110**				

Bit	Attr	Default	Description
31:20	RO-V	0x148	next_capability_offset: This field points to the next Capability in extended configuration space. In PCIe Mode, it points to the Advanced Error Capability.
19:16	RO	0x1	capability_version: Set to 1h for this version of the PCI Express* logic
15:0	RO	0xd	pci_express_extended_cap_id: Assigned for Access Control Services capability by PCISIG.

7.2.65 acscap

Access Control Services Capability Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x114**				

Bit	Attr	Default	Description
15:8	RO	0x0	egress_control_vector_size: N/A for IIO
6:6	RO	0x0	t: Applies only to root ports. Indicates that the component does not implement ACS Direct Translated P2P.
5:5	RO	0x0	e: Applies only to root portsIndicates that the component does not implement ACS P2P Egress Control.
4:4	RO-V (Device 2 and 3 Function 0) RO (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x1	u: Applies only to root ports. Indicates that the component implements ACS Upstream Forwarding.
3:3	RO-V (Device 2 and 3 Function 0) RO (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x1	c: Applies only to root ports. Indicates that the component implements ACS P2P Completion Redirect.
2:2	RO-V (Device 2 and 3 Function 0) RO (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x1	r: Applies only to root ports. Indicates that the component implements ACS P2P Request Redirect.
1:1	RO-V (Device 2 and 3 Function 0) RO (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x1	b: Applies only to root ports Indicates that the component implements ACS Translation Blocking.
0:0	RO-V (Device 2 and 3 Function 0) RO (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x1	v: Applies only to root ports Indicates that the component implements ACS Source Validation.

7.2.66 acsctrl

Access Control Services Control Register.

Type: CFG **Bus:** 0 **Bus:** 0 **Bus:** 0 **Offset:** 0x116	**PortID:** N/A **Device:** 0 **Device:** 2 **Device:** 3		 **Function:** 0 (PCIe Mode) **Function:** 0-3 **Function:** 0-3

Bit	Attr	Default	Description
6:6	RO	0x0	t: Applies only to root ports. This is hardwired to 0b as the component does not implement ACS Direct Translated P2P.
5:5	RO	0x0	e: Applies only to root ports. The component does not implement ACS P2P Egress Control and hence this bit should not be used by software.
4:4	RW-L (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	u: When this bit is set, transactions arriving from a root port that target the same port back down, will be forwarded. Normally such traffic would be aborted. Applies only to root ports.
3:3	RW-L (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	c: Applies only to root ports. Determines when the component redirects peer-to-peer Completions upstream; applicable only to Read Completions whose Relaxed Ordering Attribute is clear.
2:2	RW-L (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	r: When this bit is set, transactions arriving from a root port that target the same port back down, will be forwarded. Normally such traffic would be aborted. Applies only to root ports.
1:1	RW-L (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	b: Applies only to root ports When set, the component blocks all upstream Memory Requests whose Address Translation AT field is not set to the default value.
0:0	RW-L (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	v: Applies only to root ports. When set, the component validates the Bus Number from the Requester ID of upstream Requests against the secondary subordinate Bus Numbers.

7.2.67 apicbase

ACPI Base Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x140**				

Bit	Attr	Default	Description
11:1	RW	0x0	addr: Bits 31:20 are assumed to be 0xFECh. Bits 8:0 are a don't care for address decode. Address decoding to the APIC range is done as APICBASE.ADDR[31:8] <= A[31:8] <= APICLIMIT.ADDR[31:8]. Outbound accesses to the APIC range are claimed by the root port and forwarded to PCIe, if bit 0 is set, even if the MSE bit of the root port is clear or the root port itself is in D3hot state.
0:0	RW	0x0	en: enables the decode of the APIC window

7.2.68 apiclimit

ACPI Limit Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x142**				

Bit	Attr	Default	Description
11:1	RW	0x0	addr: Applies only to root ports. Bits 31:20 are assumed to be 0xFECh. Bits 8:0 are a don't care for address decode. Address decoding to the APIC range is done as APICBASE.ADDR[31:8] <= A[31:8] <= APICLIMIT.ADDR[31:8]. Outbound accesses to the APIC range are claimed by the root port and forwarded to PCIe, if the range is enabled, even if the MSE bit of the root port is clear or the root port itself is in D3hot state.

7.2.69 vsecphdr

PCI Express* Enhanced Capability Header - DMI2 Mode.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x144**				

Bit	Attr	Default	Description
31:20	RO	0x1d0	next_capability_offset: This field points to the next Capability in extended configuration space or is 0 if it is that last capability.
19:16	RO	0x1	capability_version: Set to 1h for this version of the PCI Express* logic
15:0	RO	0xb	pci_express_extended_cap_id: Assigned for Vendor Specific Capability

7.2.70 vshdr

Vendor Specific Header - DMI2 Mode.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**3**	**Function:**	**0 (DMI2 Mode)**
Offset:	**0x148**				

Bit	Attr	Default	Description
31:20	RO	0x3c	vsec_length: This field points to the next Capability in extended configuration space which is the ACS capability at 150h.
19:16	RO	0x1	vsec_version: Set to 1h for this version of the PCI Express* logic
15:0	RO	0x4	vsec_id: Identifies Intel Vendor Specific Capability for AER on DMI

7.2.71 errcaphdr

PCI Express* Enhanced Capability Header - Root Ports.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0 (PCIe Mode)**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x148**				

Bit	Attr	Default	Description
31:20	RO	0x1d0	next_capability_offset: This field points to the next Capability in extended configuration space or is 0 if it is that last capability.
19:16	RO	0x1	capability_version: Set to 1h for this version of the PCI Express* logic
15:0	RO	0x1	pci_express_extended_cap_id: Assigned for advanced error reporting

7.2.72 uncerrsts

Uncorrectable Error Status.

This register identifies uncorrectable errors detected for PCI Express*/DMI port.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x14c**				

Bit	Attr	Default	Description
21:21	RW1CS	0x0	acs_violation_status:
20:20	RW1CS	0x0	received_an_unsupported_request:
18:18	RW1CS	0x0	malformed_tlp_status:
17:17	RW1CS	0x0	receiver_buffer_overflow_status:
16:16	RW1CS	0x0	unexpected_completion_status:
15:15	RW1CS	0x0	completer_abort_status:
14:14	RW1CS	0x0	completion_time_out_status:
13:13	RW1CS	0x0	flow_control_protocol_error_status:
12:12	RW1CS	0x0	poisoned_tlp_status:
5:5	RW1CS	0x0	surprise_down_error_status:
4:4	RW1CS	0x0	data_link_protocol_error_status:

7.2.73 uncerrmsk

Uncorrectable Error Mask.

This register masks uncorrectable errors from being signaled.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x150**				

Bit	Attr	Default	Description
21:21	RWS	0x0	acs_violation_mask:
20:20	RWS	0x0	unsupported_request_error_mask:
18:18	RWS	0x0	malformed_tlp_mask:
17:17	RWS	0x0	receiver_buffer_overflow_mask:
16:16	RWS	0x0	unexpected_completion_mask:
15:15	RWS	0x0	completer_abort_mask:
14:14	RWS	0x0	completion_time_out_mask:
13:13	RWS	0x0	flow_control_protocol_error_mask:
12:12	RWS	0x0	poisoned_tlp_mask:
5:5	RWS	0x0	surprise_down_error_mask:
4:4	RWS	0x0	data_link_layer_protocol_error_mask:

7.2.74 uncerrsev

Uncorrectable Error Severity.

This register indicates the severity of the uncorrectable errors.

Type: CFG	**PortID:** N/A		
Bus: 0	**Device:** 0	**Function:** 0	
Bus: 0	**Device:** 2	**Function:** 0-3	
Bus: 0	**Device:** 3	**Function:** 0-3	
Offset: 0x154			

Bit	Attr	Default	Description
21:21	RWS	0x0	acs_violation_severity:
20:20	RWS	0x0	unsupported_request_error_severity:
18:18	RWS	0x1	malformed_tlp_severity:
17:17	RWS	0x1	receiver_buffer_overflow_severity:
16:16	RWS	0x0	unexpected_completion_severity:
15:15	RWS	0x0	completer_abort_severity:
14:14	RWS	0x0	completion_time_out_severity:
13:13	RWS	0x1	flow_control_protocol_error_severity:
12:12	RWS	0x0	poisoned_tlp_severity:
5:5	RWS	0x1	surprise_down_error_severity:
4:4	RWS	0x1	data_link_protocol_error_severity:

7.2.75 corerrsts

Correctable Error Status.

This register identifies the status of the correctable errors that have been detected by the PCI Express* port.

Type: CFG	**PortID:** N/A		
Bus: 0	**Device:** 0	**Function:** 0	
Bus: 0	**Device:** 2	**Function:** 0-3	
Bus: 0	**Device:** 3	**Function:** 0-3	
Offset: 0x158			

Bit	Attr	Default	Description
13:13	RW1CS	0x0	advisory_non_fatal_error_status:
12:12	RW1CS	0x0	replay_timer_time_out_status:
8:8	RW1CS	0x0	replay_num_rollover_status:
7:7	RW1CS	0x0	bad_dllp_status:
6:6	RW1CS	0x0	bad_tlp_status:
0:0	RW1CS	0x0	receiver_error_status:

7.2.76 corerrmsk

Correctable Error Mask.

This register masks correctable errors from being signaled.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x15c**				

Bit	Attr	Default	Description
13:13	RWS	0x1	advisory_non_fatal_error_mask:
12:12	RWS	0x0	replay_timer_time_out_mask:
8:8	RWS	0x0	replay_num_rollover_mask:
7:7	RWS	0x0	bad_dllp_mask:
6:6	RWS	0x0	bad_tlp_mask:
0:0	RWS	0x0	receiver_error_mask:

7.2.77 errcap

Advanced Error capabilities and Control Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x160**				

Bit	Attr	Default	Description
8:8	RO	0x0	ecrc_check_enable: N/A for IIO.
7:7	RO	0x0	ecrc_check_capable: N/A for IIO.
6:6	RO	0x0	ecrc_generation_enable: N/A for IIO.
5:5	RO	0x0	ecrc_generation_capable: N/A for IIO.
4:0	ROS_V	0x0	first_error_pointer: The First Error Pointer is a read-only register that identifies the bit position of the first unmasked error reported in the Uncorrectable Error register. In case of two errors happening at the same time, fatal error gets precedence over non-fatal, in terms of being reported as first error. This field is rearmed to capture new errors when the status bit indicated by this field is cleared by software.

7.2.78 hdrlog[0:3]

Header Log 0-3.

This register contains the header log when the first error occurs. Headers of the subsequent errors are not logged.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x164, 0x168, 0x16c, 0x170**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	hdr: Logs the first Dword of the header on an error condition.

7.2.79 rperrcmd

Root Port Error Command.

This register controls behavior upon detection of errors.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x174**				

Bit	Attr	Default	Description
2:2	RW	0x0	fatal_error_reporting_enable: Applies to root ports only. Enable MSIINTx interrupt on fatal errors when set.
1:1	RW	0x0	non_fatal_error_reporting_enable: Applies to root ports only. Enable interrupt on a non-fatal error when set.
0:0	RW	0x0	correctable_error_reporting_enable: Applies to root ports only. Enable interrupt on correctable errors when set.

7.2.80 rperrsts

Root Port Error Status.

The Root Error Status register reports status of error Messages (ERR_COR), ERR_NONFATAL, and ERR_FATAL) received by the Root Complex in IIO, and errors detected by the Root Port itself (which are treated conceptually as if the Root Port had sent an error Message to itself). The ERR_NONFATAL and ERR_FATAL Messages are grouped together as uncorrectable. Each correctable and uncorrectable (Non-fatal and Fatal) error source has a first error bit and a next error bit associated with it respectively. When an error is received by a Root Complex, the respective first error bit is set and the Requestor ID is logged in the Error Source Identification register. A set individual error status bit indicates that a particular error category occurred; software may clear an error status by writing a 1 to the respective bit. If software does not clear the first reported error before another error Message is received of the same category (correctable or uncorrectable), the corresponding next error status bit will be set but the Requestor ID of the subsequent error Message is discarded. The next error status bits may be cleared by software by writing a 1 to the respective bit as well.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x178**				

Bit	Attr	Default	Description
31:27	RO	0x0	advanced_error_interrupt_message_number: Advanced Error Interrupt Message Number offset between base message data an the MSI message if assigned more than one message number. IIO hardware automatically updates this register to 0x1h if the number of messages allocated to the root port is 2.
6:6	RW1CS	0x0	fatal_error_messages_received: Set when one or more Fatal Uncorrectable error Messages have been received.
5:5	RW1CS	0x0	non_fatal_error_messages_received: Set when one or more Non-Fatal Uncorrectable error Messages have been received.
4:4	RW1CS	0x0	first_uncorrectable_fatal: Set when bit 2 is set (from being clear) and the message causing bit 2 to be set is an ERR_FATAL message.
3:3	RW1CS	0x0	multiple_error_fatal_nonfatal_received: Set when either a fatal or a non-fatal error message is received and Error Fatal/Nonfatal Received is already set, that is, log from the 2nd Fatal or No fatal error message onwards.
2:2	RW1CS	0x0	error_fatal_nonfatal_received: Set when either a fatal or a non-fatal error message is received and this bit is already not set. that is, log the first error message. **Note:** When this bit is set bit 3 could be either set or clear.
1:1	RW1CS	0x0	multiple_correctable_error_received: Set when either a correctable error message is received and Correctable Error Received bit is already set, that is, log from the 2nd Correctable error message onwards.
0:0	RW1CS	0x0	correctable_error_received: Set when a correctable error message is received and this bit is already not set, that is, log the first error message.

7.2.81 errsid

Error Source Identification.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x17c**				

Bit	Attr	Default	Description
31:16	ROS_V	0x0	fatal_non_fatal_error_source_id: Requestor ID of the source when an Fatal or Non Fatal error message is received and the Error Fatal/Nonfatal Received bit is not already set, that is, log ID of the first Fatal or Non Fatal error message. **Note:** When the root port itself is the cause of the received message (virtual message), then a Source ID of CPUBUSNO0:DevNo:0 is logged into this register.
15:0	ROS_V	0x0	correctable_error_source_id: Requestor ID of the source when a correctable error message is received and the Correctable Error Received bit is not already set, that is, log ID of the first correctable error message. **Note:** When the root port itself is the cause of the received message (virtual message), then a Source ID of CPUBUSNO0:DevNo:0 is logged into this register.

7.2.82 perfctrlsts_0

Performance Control and Status Register 0

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x180**				

Bit	Attr	Default	Description
20:16	RW	0x18	outstanding_requests_gen1:
13:8	RW	0x30	outstanding_requests_gen2:
7:7	RW	0x1	use_allocating_flow_wr: Use Allocating Flows for 'Normal Writes' on VC0 and VCp 1: Use allocating flows for the writes that meet the following criteria. 0: Use non-allocating flows for writes that meet the following criteria. (TPH=0 OR TPHDIS=1 OR (TPH=1 AND Tag=0 AND CIPCTRL[28]=1)) AND (NS=0 OR NoSnoopOpWrEn=0) AND Non-DCA Write **Note:** VC1/VCm traffic is not impacted by this bit in Dev#0 When allocating flows are used for the above write types, IIO does not send a Prefetch Hint message. Current recommendation for the BIOS is to just leave this bit at default of 1b for all but DMI port. For DMI port when operating in DMI mode, this bit must be left at default value and when operating in PCIe mode, this bit should be set by the BIOS. **Note:** There is a coupling between the usage of this bit and bits 2 and 3. TPHDIS is bit 0 of this register NoSnoopOpWrEn is bit 3 of this register
6:6	RW	0x0	vcp_roen_nswr: Only available for Device 0 Function 0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x180**				

Bit	Attr	Default	Description
5:5	RW	0x0	vcp_nsen_rd: Only available for Device 0 Function 0.
4:4	RW	0x1	read_stream_interleave_size:
3:3	RW	0x0	nosnoopopwren: Enable No-Snoop Optimization on VC0 writes and VCp writes This applies to writes with the following conditions: NS=1 AND (TPH=0 OR TPHDIS=1) 1: Inbound writes to memory with above conditions will be treated as non-coherent (no snoops) writes 0: Inbound writes to memory with above conditions will be treated as allocating or non-allocating writes, depending on bit 4 in this register. If TPH=1 and TPHDIS=0 then NS is ignored and this bit is ignored VC1/VCm writes are not controlled by this bit since they are always non-snoop and can be no other way. Current recommendation for the BIOS is to just leave this bit at default of 0b.
2:2	RW	0x0	nosnoopoprden: Enable No-Snoop Optimization on VC0 reads and VCp reads This applies to reads with the following conditions: NS=1 AND (TPH=0 OR TPHDIS=1) 1: When the condition is true for a given inbound read request to memory, it will be treated as non-coherent (no snoops) reads. 0: When the condition is true for a given inbound read request to memory, it will be treated as normal snooped reads from PCIe (which trigger a PCIRdCurrent or DRd.UC on IDI). **Notes:** • If TPH=1 and TPHDIS=0 then NS is ignored and this bit is ignored • VC1 and VCm reads are not controlled by this bit and those reads are always non-snoop. • Current recommendation for the BIOS is to just leave this bit at default of 0b.
1:1	RW	0x0	read_passing_read_disable:
0:0	RW	0x1	read_stream_policy:

7.2.83 perfctrlsts_1

Performance Control and Status Register 1

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x184**				

Bit	Attr	Default	Description
9:9	RW	0x0	tphdis: TLP Processing Hint Disable When set, writes or reads with TPH=1, will be treated as if TPH=0.
8:8	RW	0x0	dca_reqid_override: DCA Requester ID Override When this bit is set, Requester ID match for DCA writes is bypassed. All writes from the port are treated as DCA writes and the tag field will convey if DCA is enabled or not and the target information.
3:3	RW	0x0	max_read_completion_combine_size:

7.2.84 miscctrlsts_0

MISC Control and Status Register 0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x188**				

Bit	Attr	Default	Description
31:31	RW	0x0	disable_l0s_on_transmitter: When set, IIO never puts its tx in L0s state, even if OS enables it by means of the Link Control register.
30:30	RW-O	0x1	inbound_io_disable:
29:29	RW	0x1	cfg_to_en: Disables/enables config timeouts, independently of other timeouts.
28:28	RW	0x0	to_dis: Disables timeouts completely.
27:27	RWS	0x0	system_interrupt_only_on_link_bw_management_status: This bit, when set, will disable generating MSI and Intx interrupts on link bandwidth (speed and/or width) and management changes, even if MSI or INTx is enabled that is, will disable generating MSI or INTx when LNKSTS bits 15 and 14 are set. Whether or not this condition results in a system event like SMI/PMI/CPEI is dependent on whether this event masked or not in the XPCORERRMSK register.
26:26	RW_LV (Device 2 and 3 Function 0) RW (Device 0 Function 0, Device 2 and 3 Function 1-3)	0x0	eoifd: EOI Forwarding Disable—Disable EOI broadcast to this PCIe link When set, EOI message will not be broadcast down this PCIe link. When clear, the port is a valid target for EOI broadcast.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x188**				

Bit	Attr	Default	Description
24:24	RW	0x0	peer2peer_memory_read_disable: When set, peer-to-peer memory reads are master aborted otherwise they are allowed to progress per the peer-to-peer decoding rules.
23:23	RW	0x0	phold_disable: Applies only to Dev#0When set, the IIO responds with Unsupported request on receiving assert_phold message from ICH and results in generating a fatal error.
22:22	RWS	0x0	check_cpl_tc:
21:21	RW-O	0x0	zero_ob_tc: Forces the TC field to zero for outbound requests. 1: TC is forced to zero on all outbound transactions regardless of the source TC value 0: TC is not altered **Note:** In DMI mode, TC is always forced to zero and this bit has no effect.
20:20	RW	0x1	maltlp_32baddr64bhdr_en: When set, enables reporting a Malformed packet when the TLP is a 32-bit address in a 4DW header. PCI Express* forbids using 4DW header sizes when the address is less than 4GB, but some cards may use the 4DW header anyway. In these cases, the upper 32 bits of address are all 0.
18:18	RWS	0x0	max_read_completion_combine_size: When set, all completions are returned without combining. Completions are naturally broken on cacheline boundaries, so all completions will be 64B or less.
17:17	RO	0x0	force_data_perr:
16:16	RO	0x0	force_ep_biterr:
15:15	RWS	0x0	dis_hdr_storage:
14:14	RWS	0x0	allow_one_np_os:
13:13	RWS	0x0	tlp_on_any_lane:
12:12	RWS	0x1	disable_ob_parity_check:
11:11	RWS	0x1	allow_1nonvc1_after_10vc1s: Allow a non-VC1 request from DMI to go after every ten VC1 request (to prevent starvation of non-VC1). Only avaiable for Device 0 Function 0.
9:9	RWS	0x0	dispdspolling: Disables gen2 if timeout happens in polling.cfg.
8:7	RW	0x0	pme2acktoctrl:
6:6	RW	0x0	enable_timeout_for_receiving_pme_to_ack: When set, IIO enables the timeout to receiving the PME_TO_ACK
5:5	RW-V	0x0	send_pme_turn_off_message: When this bit is written with a 1b, IIO sends a PME_TURN_OFF message to the PCIe link. Hardware clears this bit when the message has been sent on the link.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x188**				

Bit	Attr	Default	Description
4:4	RW	0x0	enable_system_error_only_for_aer: Applies only to root ports. For Dev#0 in DMI mode, this bit is to be left at default value always.When this bit is set, the PCI Express* errors do not trigger an MSI or Intx interrupt, regardless of the whether MSI or INTx is enabled or not. Whether or not PCI Express* errors result in a system event like NMI/SMI/PMI/CPEI is dependent on whether the appropriate system error or override system error enable bits are set or not. When this bit is clear, PCI Express* errors are reported by means of MSI or INTx and/or NMI/SMI/MCA/CPEI. When this bit is clear, and 'System Error on Fatal Error Enable' bit in ROOTCON register is set, then NMI/SMI/MCA is (also) generated for a PCI Express* fatal error. Similar behavior for non-fatal and corrected errors.
3:3	RW	0x0	enable_acpi_mode_for_hotplug:
2:2	RW	0x0	enable_acpi_mode_for_pm:
1:1	RW-O	0x0	inbound_configuration_enable:

7.2.85 miscctrlsts_1

MISC Control and Status Register 1.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x18c**				

Bit	Attr	Default	Description
19:19	RW	0x1	vcm_arb_in_vc1: Only available for Device 0 Function 0.
18:18	RW	0x0	no_vcm_throttle_in_quiesce: Only available for Device 0 Function 0
17:17	RW1CS	0x0	locked_read_timed_out: Indicates that a locked read request incurred a completion time-out on PCI Express*/DMI
16:16	RW1C	0x0	received_pme_to_ack: Indicates that IIO received a PME turn off ack packet or it timed out waiting for the packet
9:9	RW	0x0	override_socketid_in_cplid: For TPH/DCA requests, the Completer ID can be returned with SocketID when this bit is set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x18c**				

Bit	Attr	Default	Description
6:6	RW	0x0	problematic_port_for_lock_flows: This bit is set by the BIOS when it knows that this port is connected to a device that creates Posted-Posted dependency on its In-Out queues. Briefly, this bit is set on a link if: IIO lock flows depend on the setting of this bit to treat this port in a special way during the flows. **Note:** An inbound MSI request can block the posted channel until EOI's are posted to all outbound queues enabled to receive EOI. Because of this, this bit cannot be set unless EOIFD is also set.
5:5	RW	0x0	disable_mctp_broadcast_to_this_link: When set, this bit will prevent a broadcast MCTP message (w/ Routing Type of 'Broadcast from RC') from being sent to this link.
4:4	RWS	0x0	formfactor: Indicates what form-factor a particular root port controls 0 - CEM 1 - Express* Module This bit is used to interpret bit 6 in the VPP serial stream for the port as either MRL# (CEM) input or EMLSTS# (Express Module) input.
3:3	RW	0x0	override_system_error_on_pcie_fatal_error_enable: When set, fatal errors on PCI Express* (that have been successfully propagated to the primary interface of the port) are sent to the IIO core error logic (for further escalation) regardless of the setting of the equivalent bit in the ROOTCTRL register. When clear, the fatal errors are only propagated to the IIO core error logic if the equivalent bit in ROOTCTRL register is set. For Device 0 in DMI mode and Dev#3/Fn#0, unless this bit is set, DMI link related fatal errors will never be notified to system software.
2:2	RW	0x0	override_system_error_on_pcie_non_fatal_error_enable: When set, non-fatal errors on PCI Express* (that have been successfully propagated to the primary interface of the port) are sent to the IIO core error logic (for further escalation) regardless of the setting of the equivalent bit in the ROOTCTRL register. When clear, the non-fatal errors are only propagated to the IIO core error logic if the equivalent bit in ROOTCTRL register is set. For Dev#0 in DMI mode and Dev#3/Fn#0, unless this bit is set, DMI link related non-fatal errors will never be notified to system software.
1:1	RW	0x0	override_system_error_on_pcie_correctable_error_enable: When set, correctable errors on PCI Express* (that have been successfully propagated to the primary interface of the port) are sent to the IIO core error logic (for further escalation) regardless of the setting of the equivalent bit in the ROOTCTRL register. When clear, the correctable errors are only propagated to the IIO core error logic if the equivalent bit in ROOTCTRL register is set. For Dev#0 in DMI mode and Dev#3/Fn#0, unless this bit is set, DMI link related correctable errors will never be notified to system software.
0:0	RW	0x0	acpi_pme_inten: When set, Assert/Deassert_PMEGPE messages are enabled to be generated when ACPI mode is enabled for handling PME messages from PCI Express*. When this bit is cleared (from a 1), a Deassert_PMEGPE message is scheduled on behalf of the root port if an Assert_PMEGPE message was sent last from the root port.

7.2.86 pcie_iou_bif_ctrl

PCIe Port Bifurcation Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x190**				

Bit	Attr	Default	Description
3:3	WO	0x0	iou_start_bifurcation: When software writes a 1 to this bit, IIO starts the port 0 bifurcation process. After writing to this bit, software can poll the Data Link Layer link active bit in the LNKSTS register to determine if a port is up and running. Once a port bifurcation has been initiated by writing a 1 to this bit, software cannot initiate any more write-1 to this bit (write of 0 is ok). **Notes:** • That this bit can be written to a 1 in the same write that changes values for bits 2:0 in this register and in that case, the new value from the write to bits 2:0 take effect. • This bit always reads a 0b.
2:0	RWS RO (Device 0 Function 0)	0x4 0x0 (Device 0 Function 0)	iou_bifurcation_control: To select a IOU bifurcation, software sets this field and then either a) sets bit 3 in this register to initiate training OR b) resets the entire the processor and on exit from that reset, the processor will bifurcate the ports per the setting in this field. For Device 2 and Device 3 Function 0: 000: x4x4x4x4 operate lanes 15:12 as x4, 11:8 as x4, 7:4 as x4 and 3:0 as x4 001: x4x4x8 operate lanes 15:12 as x4, 11:8 as x4 and 7:0 as x8 010: x8x4x4 operate lanes 15:8 as x8, 7:4 as x4 and 3:0 as x4 011: x8x8 operate lanes 15:8 as x8, 7:0 as x8 100: x16 others: Reserved For Device 0 Function 0, read only.

7.2.87 dmictrl

Type: CFG **Bus:** 0 **Offset:** 0x1a0	**PortID:** N/A **Device:** 0	**Function:** 0 (DMI2 Mode)	

Bit	Attr	Default	Description
63:2	RO	0x0	rsvd:
1:1	RW	0x1	auto_complete_pm: This bit, if set, enables the DMI port to automatically complete PM message handshakes by generating an AckSx or RstWarnAck message down DMI for the following DMI messages received: GoS0 GoS1RW GoS1Temp GoS1Final GoS3 GoS4 GoS5 RstWarn **Note:** This is used by pCode to indicate periods of time when it is not ready to accept messages and there is a risk the messages will be lost.
0:0	RW	0x1	abort_inbound_requests: Setting this bit causes the IIO to abort all inbound requests on the DMI port. This is used during specific power state and reset transitions to prevent requests from the PCH. This bit does not apply in PCI Express mode. Inbound posted requests will be dropped and inbound non-posted requests will be completed with Unsupported Request completion. Completions flowing inbound (from outbound requests) will not be dropped, but will be forwarded normally. This bit will not affect S-state auto-completion, if it is enabled.

7.2.88 dmists

Type: CFG **Bus:** 0 **Offset:** 0x1a8	**PortID:** N/A **Device:** 0	**Function:** 0 (DMI2 Mode)	

Bit	Attr	Default	Description
31:1	RO	0x0	reserved:
0:0	RW1C	0x0	received_cpu_reset_done_ack:

7.2.89 ERRINJCAP

PCI Express* Error Injection Capability.

Defines a vendor specific capability for WHEA error injection.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1d0**				

Bit	Attr	Default	Description
31:20	RO	0x250 0x280 (Device 0 Function 0)	nxtptr: Next Capability Offset This field points to the next capability or 0 if there isn't a next capability.
19:16	RO	0x1	capver: Capability Version Set to 2h for this version of the PCI Express* specification
15:0	RO	0xb	extcapid: PCI Express* Extended Capability ID Vendor Defined Capability

7.2.90 ERRINJHDR

PCI Express* Error Injection Capability Header.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1d4**				

Bit	Attr	Default	Description
31:20	RO	0xa	vseclen: Vendor Specific Capability Length Indicates the length of the capability structure, including header bytes.
19:16	RO	0x1	vsecrev: Vendor Specific Capability Revision Set to 1h for this version of the WHEA Error Injection logic.
15:0	RO	0x3	vsecid: Vendor Specific ID Assigned for WHEA Error Injection

7.2.91 ERRINJCON

PCI Express* Error Injection Control Register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1d8**				

Bit	Attr	Default	Description
2:2	RW	0x0	cause_ctoerr: Cause a Completion Timeout Error When this bit is written to transition from 0 to 1, one and only one error assertion pulse is produced on the error source signal for the given port. This error will appear equivalent to an actual error assertion because this event is OR'd into the existing error reporting structure. To log another error, this bit must be cleared first, before setting again. Leaving this bit in a 1 state does not produce a persistent error condition. **Notes**: • This bit is used for an uncorrectable error test • This bit must be cleared by software before creating another event. • This bit is disabled by bit 0 of this register
1:1	RW	0x0	cause_rcverr: Cause a Receiver Error When this bit is written to transition from 0 to 1, one and only one error assertion pulse is produced on the error source signal for the given port. This error will appear equivalent to an actual error assertion because this event is OR'd into the existing error reporting structure. To log another error, this bit must be cleared first, before setting again. Leaving this bit in a 1 state does not produce a persistent error condition. **Notes:** • This bit is used for an correctable error test • This bit must be cleared by software before creating another event. • This bit is disabled by bit 0 of this register
0:0	RW-O	0x0	errinjdis: Error Injection Disable This bit disables the use of the PCIe error injection bits. **Note:** This is a write once bit.

7.2.92 ctoctrl

Completion Timeout Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x1e0**				

Bit	Attr	Default	Description
9:8	RW	0x0	xp_to_pcie_timeout_select: When OS selects a timeout range of 17s to 64s for XP (that affect NP tx issued to the PCIe/DMI) using the root port's DEVCTRL2 register, this field selects the sub-range within that larger range, for additional controllability. 00 : 17s-30s 01 : 31s-45s 10 : 46s-64s 11 : Reserved

7.2.93 xpcorerrsts

XP Correctable Error Status

The contents of the next set of registers - XPCORERRSTS, XPCORERRMSK, XPUNCERRSTS, XPUNCERRMSK, XPUNCERRSEV, XPUNCERRPTR - to be defined by the design team based on microarchitecture. The architecture model for error logging and escalation of internal errors is similar to that of PCI Express* AER, except that these internal errors never trigger an MSI and are always reported to the system software. Mask bits mask the reporting of an error and severity bit controls escalation to either fatal or non-fatal error to the internal core error logic.

Note: Internal errors detected in the PCI Express* cluster are not dependent on any other control bits for error escalation other than the mask bit defined in these registers. All these registers are sticky.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x200**				

Bit	Attr	Default	Description
0:0	RW1CS	0x0	pci_link_bandwidth_changed_status: This bit is set when the logical OR of LNKSTS[15] and LNKSTS[14] goes from 0 to 1.

7.2.94 xpcorerrmsk

XP Correctable Error Mask.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x204**				

Bit	Attr	Default	Description
0:0	RWS	0x0	pci_link_bandwidth_changed_mask: Masks the BW change event from being propagated to the IIO core error logic as a correctable error

7.2.95 xpuncerrsts

XP Uncorrectable Error Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x208**				

Bit	Attr	Default	Description
9:9	RW1CS	0x0	outbound_poisoned_data: Set when outbound poisoned data is received by this port
8:8	RW1CS	0x0	received_msi_writes_greater_than_a_dword_data:
7:7	RW1CS	0x0	unused7:
6:6	RW1CS	0x0	received_pcie_completion_with_ur_status:
5:5	RW1CS	0x0	received_pcie_completion_with_ca_status:
4:4	RW1CS	0x0	sent_completion_with_unsupported_request:
3:3	RW1CS	0x0	sent_completion_with_completer_abort:
2:2	RW1CS	0x0	unused2:
1:1	RW1CS	0x0	outbound_switch_fifo_data_parity_error_detected:
0:0	RW1CS	0x0	unused0:

7.2.96 xpuncerrmsk

XP Uncorrectable Error Mask.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x20c**				

Bit	Attr	Default	Description
9:9	RWS	0x0	outbound_poisoned_data_mask: Masks signaling of stop and scream condition to the core error logic.
8:8	RWS	0x0	received_msi_writes_greater_than_a_dword_data_mask:
7:7	RWS	0x0	unused7:
6:6	RWS	0x0	received_pcie_completion_with_ur_status_mask:
5:5	RWS	0x0	received_pcie_completion_with_ca_status_mask:
4:4	RWS	0x0	sent_completion_with_unsupported_request_mask:
3:3	RWS	0x0	sent_completion_with_completer_abort_mask:
2:2	RWS	0x0	unused2:
1:1	RWS	0x0	outbound_switch_fifo_data_parity_error_detected_mask:
0:0	RWS	0x0	unused0:

7.2.97 xpuncerrsev

XP Uncorrectable Error Severity

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x210**				

Bit	Attr	Default	Description
9:9	RWS	0x0	outbound_poisoned_data_severity:
8:8	RWS	0x0	received_msi_writes_greater_than_a_dword_data_severity:
7:7	RWS	0x0	unused7:
6:6	RWS	0x0	received_pcie_completion_with_ur_status_severity:
5:5	RWS	0x0	received_pcie_completion_with_ca_status_severity:
4:4	RWS	0x0	sent_completion_with_unsupported_request_severity:
3:3	RWS	0x0	sent_completion_with_completer_abort_severity:
2:2	RWS	0x0	unused2:
1:1	RWS	0x1	outbound_switch_fifo_data_parity_error_detected_severity:
0:0	RWS	0x0	unused0:

7.2.98 xpuncerrptr

XP Uncorrectable Error Pointer.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x214**				

Bit	Attr	Default	Description
4:0	ROS_V	0x0	xp_uncorrectable_first_error_pointer: This field points to which of the unmasked uncorrectable errors happened first. This field is only valid when the corresponding error is unmasked and the status bit is set and this field is rearmed to load again when the status bit indicated to by this pointer is cleared by software from 1 to 0.Value of 0x0 corresponds to bit 0 in XPUNCERRSTS register, value of 0x1 corresponds to bit 1 and so forth.

7.2.99 uncedmask

Uncorrectable Error Detect Status Mask

This register masks PCIe link related uncorrectable errors from causing the associated AER status bit to be set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x218**				

Bit	Attr	Default	Description
21:21	RWS	0x0	acs_violation_detect_mask:
20:20	RWS	0x0	received_an_unsupported_request_detect_mask:
18:18	RWS	0x0	malformed_tlp_detect_mask:
17:17	RWS	0x0	receiver_buffer_overflow_detect_mask:
16:16	RWS	0x0	unexpected_completion_detect_mask:
15:15	RWS	0x0	completer_abort_detect_mask:
14:14	RWS	0x0	completion_time_out_detect_mask:
13:13	RWS	0x0	flow_control_protocol_error_detect_mask:
12:12	RWS	0x0	poisoned_tlp_detect_mask:
5:5	RWS	0x0	surprise_down_error_detect_mask:
4:4	RWS	0x0	data_link_layer_protocol_error_detect_mask:

7.2.100 coredmask

Correctable Error Detect Status Mask

This register masks PCIe link related correctable errors from causing the associated status bit in AER status register to be set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x21c**				

Bit	Attr	Default	Description
13:13	RWS	0x0	advisory_non_fatal_error_detect_mask:
12:12	RWS	0x0	replay_timer_time_out_detect_mask:
8:8	RWS	0x0	replay_num_rollover_detect_mask:
7:7	RWS	0x0	bad_dllp_detect_mask:
6:6	RWS	0x0	bad_tlp_detect_mask:
0:0	RWS	0x0	receiver_error_detect_mask:

7.2.101 rpedmask

Root Port Error Detect Status Mask

This register masks the associated error messages (received from PCIe link and NOT the virtual ones generated internally), from causing the associated status bits in AER to be set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x220**				

Bit	Attr	Default	Description
2:2	RWS	0x0	fatal_error_detected_status_mask:
1:1	RWS	0x0	non_fatal_error_detected_status_mask:
0:0	RWS	0x0	correctable_error_detected_status_mask:

7.2.102 xpuncedmask

XP Uncorrectable Error Detect Mask

This register masks other uncorrectable errors from causing the associated XPUNCERRSTS status bit to be set

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x224**				

Bit	Attr	Default	Description
9:9	RWS	0x0	outbound_poisoned_data_detect_mask:
8:8	RWS	0x0	received_msi_writes_greater_than_a_dword_data_detect_mask:
7:7	RWS	0x0	unused7:
6:6	RWS	0x0	received_pcie_completion_with_ur_detect_mask:
5:5	RWS	0x0	received_pcie_completion_with_ca_detect_mask:
4:4	RWS	0x0	sent_completion_with_unsupported_request_detect_mask:
3:3	RWS	0x0	sent_completion_with_completer_abort_detect_mask:
2:2	RWS	0x0	unused2:
1:1	RWS	0x0	outbound_switch_fifo_data_parity_error_detect_mask:
0:0	RWS	0x0	unused0:

7.2.103 xpcoredmask

XP Correctable Error Detect Mask

This register masks other correctable errors from causing the associated XPCORERRSTS status bit to be set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x228**				

Bit	Attr	Default	Description
0:0	RWS	0x0	pci_link_bandwidth_changed_detect_mask:

7.2.104 xpglberrsts

XP Global Error Status

This register captures a concise summary of the error logging in AER registers so that sideband system management software can view the errors independent of the main OS that might be controlling the AER errors.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x230**				

Bit	Attr	Default	Description
2:2	RW1CS	0x0	pcie_aer_correctable_error: A PCIe correctable error (ERR_COR message received from externally or through a virtual ERR_COR message generated internally) was detected anew. **Note:** If that error was masked in the PCIe AER, it is not reported in this field. Software clears this bit by writing a 1 and at that stage, only 'subsequent' PCIe unmasked correctable errors will set this bit.Conceptually, per the flow of PCI Express* Base Specification 2.0 defined Error message control, this bit is set by the ERR_COR message that is enabled to cause a System Error notification..
1:1	RW1CS	0x0	pcie_aer_non_fatal_error: A PCIe non-fatal error (ERR_NONFATAL message received from externally or through a virtual ERR_NONFATAL message generated internally) was detected anew. **Note:** If that error was masked in the PCIe AER, it is not reported in this field. Software clears this bit by writing a 1 and at that stage only 'subsequent' PCIe unmasked non-fatal errors will set this bit again.
0:0	RW1CS	0x0	pcie_aer_fatal_error: A PCIe fatal error (ERR_FATAL message received from externally or through a virtual ERR_FATAL message generated internally) was detected anew. **Note:** If that error was masked in the PCIe AER, it is not reported in this field. Software clears this bit by writing a 1 and at that stage, only 'subsequent' PCIe unmasked fatal errors will set this bit.

7.2.105 xpglberrptr

XP Global Error Pointer

Check that the perfmon registers are per “cluster”.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x232**				

Bit	Attr	Default	Description
2:0	ROS_V	0x0	xp_cluster_global_first_error_pointer: This field points to which of the 3 errors indicated in the XPGLBERRSTS register happened first. This field is only valid when the corresponding status bit is set and this field is rearmed to load again when the status bit indicated to by this pointer is cleared by software from 1 to 0.Value of 0x0 corresponds to bit 0 in XPGLBERRSTS register, value of 0x1 corresponds to bit 1, and so forth.

7.2.106 pxp2cap

Secondary PCI Express* Extended Capability Header.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x250**				

Bit	Attr	Default	Description
31:20	RO	0x280	nxtptr: Next Capability Offset. This field contains the offset to the next PCI Express* Extended Capability structure or 000h if no other items exist in the linked list of capabilities.
19:16	RW-O	0x1	version: This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.
15:0	RW-O	0x19	id: This field is a PCI SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express* Extended Capability ID for the Secondary PCI Express Extended Capability is 0019h.

7.2.107 lnkcon3

Link Control 3 Register.

Type: CFG **Bus:** 0 **Bus:** 0 **Offset:** 0x254	**PortID:** N/A **Device:** 2 **Device:** 3	**Function:** 0-3 **Function:** 0-3	
Bit	**Attr**	**Default**	**Description**
1:1	RW	0x0	lnkeqreqinten: Link Equalization Request Interrupt Enable. When Set, this bit enables the generation of interrupt to indicate that the Link Equalization Request bit has been set.
0:0	RW	0x0	perfeq: Performance Equalization. When this register is 1b and a 1b is written to the 'Link Retrain' register with 'Target Link Speed' set to 8GTs, the Upstream component must perform Transmitter Equalization.

7.2.108 lnerrsts

Lane Error Status Register

Type: CFG **Bus:** 0 **Bus:** 0 **Offset:** 0x258	**PortID:** N/A **Device:** 2 **Device:** 3	**Function:** 0-3 **Function:** 0-3	
Bit	**Attr**	**Default**	**Description**
15:0	RW1CS	0x0	lane: A value of 1b in any bit indicates if the corresponding PCIe Express* Lane detected lane based error. bit 0 Lane 0 Error Detected bit 1 Lane 1 Error Detected bit 2 Lane 2 Error Detected bit 3 Lane 3 Error Detected bit 4 Lane 4 Error Detected (not used when the link is bifurcated as x4) bit 5 Lane 5 Error Detected (not used when the link is bifurcated as x4) bit 6 Lane 6 Error Detected (not used when the link is bifurcated as x4) bit 7 Lane 7 Error Detected (not used when the link is bifurcated as x4) bit 8 Lane 8 Error Detected (not used when the link is bifurcated as x4 or x8) bit 9 Lane 9 Error Detected (not used when the link is bifurcated as x4 or x8) bit 10 Lane 10 Error Detected (not used when the link is bifurcated as x4 or x8) bit 11 Lane 11 Error Detected (not used when the link is bifurcated as x4 or x8) bit 12 Lane 12 Error Detected (not used when the link is bifurcated as x4 or x8) bit 13 Lane 13 Error Detected (not used when the link is bifurcated as x4 or x8) bit 14 Lane 14 Error Detected (not used when the link is bifurcated as x4 or x8) bit 15 Lane 15 Error Detected (not used when the link is bifurcated as x4 or x8)

7.2.109 ln[0:3]eq

Lane 0 through Lane 3 Equalization Control

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x25c, 0x25e, 0x260, 0x262**				

Bit	Attr	Default	Description
14:12	RW-O	0x7	dnrxpreset: Downstream Component Receiver Preset Hint Receiver Preset Hint for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: Reserved For a Downstream Component, this field reflects the latest Receiver Preset value requested from the Upstream Component on Lane 0. The default value is 111b.
11:8	RW-O	0x8	dntxpreset: Downstream Component Transmitter Preset Transmitter Preset for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved For a Downstream Component, this field reflects the latest Transmitter Preset requested from the Upstream Component on Lane 0. The default value is 111b.
6:4	RO	0x7	uprxpreset: Upstream Component Receiver Preset Hint Receiver Preset Hint for Upstream Component. The upstream component uses this hint for receiver equalization. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: reserved

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x25c, 0x25e, 0x260, 0x262**				

Bit	Attr	Default	Description
3:0	RW-O	0x8	uptxpreset: Upstream Component Transmitter Preset Transmitter Preset for an Upstream Component. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved

7.2.110 ln[4:7]eq

Lane 4 through Lane 7 Equalization Control

This register is unused when the link is configured at x4 in the bifurcation register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0, 2**
Bus:	**0**	**Device:**	**3**	**Function:**	**0, 2**
Offset:	**0x264, 0x266, 0x268, 0x26a**				

Bit	Attr	Default	Description
14:12	RW-O	0x2	dnrxpreset: Downstream Component Receiver Preset Hint Receiver Preset Hint for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: Reserved For a Downstream Component, this field reflects the latest Receiver Preset value requested from the Upstream Component on Lane 0. The default value is 111b.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0, 2**
Bus:	**0**	**Device:**	**3**	**Function:**	**0, 2**
Offset:	**0x264, 0x266, 0x268, 0x26a**				

Bit	Attr	Default	Description
11:8	RW-O	0x8	dntxpreset: Downstream Component Transmitter Preset Transmitter Preset for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved For a Downstream Component, this field reflects the latest Transmitter Preset requested from the Upstream Component on Lane 0. The default value is 111b.
6:4	RO	0x7	uprxpreset: Upstream Component Receiver Preset Hint Receiver Preset Hint for Upstream Component. The upstream component uses this hint for receiver equalization. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: reserved
3:0	RW-O	0x8	uptxpreset: Upstream Component Transmitter Preset Transmitter Preset for an Upstream Component. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved

7.2.111 ln[8:15]eq

Lane 8 though Lane 15 Equalization Control

This register is unused when the link is configured at x4 or x8 in the bifurcation register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x26c, 0x26e, 0x270, 0x272, 0x274, 0x276, 0x278, 0x27a**				

Bit	Attr	Default	Description
14:12	RW-O	0x7	dnrxpreset: Downstream Component Receiver Preset Hint Receiver Preset Hint for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: Reserved For a Downstream Component, this field reflects the latest Receiver Preset value requested from the Upstream Component on Lane 0. The default value is 111b.
11:8	RW-O	0x8	dntxpreset: Downstream Component Transmitter Preset Transmitter Preset for Downstream Component with the following encoding. The Upstream component must pass on this value in the EQ TS2'es. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved For a Downstream Component, this field reflects the latest Transmitter Preset requested from the Upstream Component on Lane 0. The default value is 111b.
6:4	RO	0x7	uprxpreset: Upstream Component Receiver Preset Hint Receiver Preset Hint for Upstream Component. The upstream component uses this hint for receiver equalization. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: reserved

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x26c, 0x26e, 0x270, 0x272, 0x274, 0x276, 0x278, 0x27a**				

Bit	Attr	Default	Description
3:0	RW-O	0x8	uptxpreset: Upstream Component Transmitter Preset Transmitter Preset for an Upstream Component. The Root Ports are upstream components. The encodings are defined below. 000b: -6 dB for de-emphasis, 0 dB for preshoot 001b: -3.5 dB for de-emphasis, 0 dB for preshoot 010b: -6 dB for de-emphasis, -3.5 dB for preshoot 011b: -3.5 dB for de-emphasis, -3.5 dB for preshoot 100b: -0 dB for de-emphasis, 0 dB for preshoot 101b: -0 dB for de-emphasis, -3.5 dB for preshoot others: reserved

7.2.112 ler_cap

Live Error Recovery Capability.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x280**				

Bit	Attr	Default	Description
31:20	RO	0x0	nxtptr: Next Capability Offset.This field points to the next Capability in extended configuration space.
19:16	RO	0x1	capver: Capability Version. Set to 1h for this version of the PCI Express* logic.
15:0	RO	0xb	capid: PCI Express* Extended CAP_ID. Assigned for advanced error reporting.

7.2.113 ler_hdr

Live Error Recovery Capability Header

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**1**	**Function:**	**0-1**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x284**				

Bit	Attr	Default	Description
31:20	RO	0x18	vseclen: VSEC Length. This field indicates the length of the LER capability in bytes. It includes the capability headers.
19:16	RO	0x3	vsecrev: VSEC revision. Set to 2h for this version of the Live Error Recovery logic.
15:0	RO	0x5	vsecid: Vendor Specific ID. Assigned for Live Error Recovery.

7.2.114 ler_ctrlsts

Live Error Recovery Control and Status: LER is nor supported, only Stop and Scream

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0-3**
Bus:	**0**	**Device:**	**3**	**Function:**	**0-3**
Offset:	**0x288**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	ler_ss_status: Indicates that an error was detected that caused the PCIe port to go into a live error recovery (LER) mode. While in LER mode, the link goes into a LinkDown "Disabled" state and all outbound transactions are aborted (including packets that may have caused the error). This bit cannot be cleared until all the associated unmasked status bits are cleared. or the corresponding LER mask bits are set. Once the unmaksed error condition are cleared, then this bit may be cleared by software wrting a '1'. The link will retrain into LinkUp state and outbound transactions will no longer be aborted. Also, inbound transactions will also no longer be blocked. A link that is forced into a LinkDown "Disabled" state due to LER does not trigger a "surprise LinkDown" error in the UNCERRSTS register. **Note:** Many PCIe cards will go into internal reset when they receive training sequences that indicate the "Disabled" state.
30:30	ROS_V	0x0	ler_ss_port_quesced: Indicates when the port has no more pending inbound or outbound packets after the port has entered LER mode. It is used by software to determine when it is safe to clear the LER_SS_Status bit to bring the port out of LER mode.
29:4	RV	0x0	Reserved:
3:3	RWS	0x0	ler_ss_inten: If set, causes an INTx or MSI interrupt from the root port (if enabled in the root port) to be generated when LER_SS_Status is set.

Type: CFG Bus: 0 Bus: 0 Bus: 0 Offset: 0x288			PortID: N/A Device: 0 Function: 0 Device: 2 Function: 0-3 Device: 3 Function: 0-3
Bit	**Attr**	**Default**	**Description**
2:2	RWS	0x0	ler_ss_drop_txn: If set, after entering LER subsequent transactions will be dropped as soon as the port configuration allows
1:1	RWS	0x0	ler_ss_severity: If set, forces the errors that trigger LER mode to be signaled as a correctable error of Severity 0. If cleared, then errors are signaled as Uncorrectable Non-Fatal Severity 1 or Uncorrectable Fatal Severity 2 as specified for the given error.
0:0	RWS	0x0	ler_ss_enable: When set, allow the LER_SS Status to assert on error. When the status bit is set, the associated root port will go into LER mode. When clear, the LER_SS_Status bit can no longer be set on an error and root port can never go into LER mode. **Note**: If this bit is cleared when the LER_SS_Status bit is already set, then clearing this bit does not clear the status bit and does not exit LER mode. To exit LER mode, the Status bit must be cleared by software.

7.2.115 ler_uncerrmsk

Live Error Recovery Uncorrectable Error Mask

This register masks uncorrectable errors from being signaled as LER events.

Type: CFG Bus: 0 Bus: 0 Bus: 0 Offset: 0x28c			PortID: N/A Device: 0 Function: 0 Device: 2 Function: 0-3 Device: 3 Function: 0-3
Bit	**Attr**	**Default**	**Description**
21:21	RWS	0x0	acs_violation_mask:
20:20	RWS	0x0	unsupported_request_error_mask:
18:18	RWS	0x0	malformed_tlp_mask:
17:17	RWS	0x0	receiver_buffer_overflow_mask:
16:16	RWS	0x0	unexpected_completion_mask:
15:15	RWS	0x0	completer_abort_mask:
14:14	RWS	0x0	completion_time_out_mask:
13:13	RWS	0x0	flow_control_protocol_error_mask:
12:12	RWS	0x0	poisoned_tlp_mask:
5:5	RWS	0x0	surprise_down_error_mask:
4:4	RWS	0x0	data_link_layer_protocol_error_mask:

7.2.116 ler_xpuncerrmsk

Live Error Recovery XP Uncorrectable Error Mask.

Type: CFG **Bus:** 0 **Bus:** 0 **Bus:** 0 **Offset:** 0x290		**PortID:** N/A **Device:** 0 **Device:** 2 **Device:** 3	**Function:** 0 **Function:** 0-3 **Function:** 0-3
Bit	**Attr**	**Default**	**Description**
9:9	RWS	0x0	outbound_poisoned_data_mask: Masks signaling of stop and scream condition to the core error logic
6:6	RWS	0x0	received_pcie_completion_with_ur_status_mask:
5:5	RWS	0x0	received_pcie_completion_with_ca_status_mask:
4:4	RWS	0x0	sent_completion_with_ur_mask:
3:3	RWS	0x0	sent_completion_with_ca_mask:

7.2.117 ler_rperrmsk

Live Error Recovery Root Port Error Mask.

Type: CFG **Bus:** 0 **Bus:** 0 **Bus:** 0 **Offset:** 0x294		**PortID:** N/A **Device:** 0 **Device:** 2 **Device:** 3	**Function:** 0 **Function:** 0-3 **Function:** 0-3
Bit	**Attr**	**Default**	**Description**
6:6	RWS	0x0	fatal_error_message_received_mask: Masks LER response to Fatal Error Messages received
5:5	RWS	0x0	non_fatal_error_message_received_mask: Masks LER response to Non-Fatal Error Messages received

7.2.118 xppmdl[0:1]

XP PM Data Low Bits

This is the performance monitor counter. This counter is reset at the beginning of a sample period unless pre-loaded with a sample value. Therefore, the counter can cause an early overflow condition with values loaded into the register.

Type: CFG **Bus:** 0 **Bus:** 0 **Bus:** 0 **Offset:** 0x480, 0x484		**PortID:** N/A **Device:** 0 **Device:** 2 **Device:** 3	**Function:** 0 **Function:** 0 **Function:** 0
Bit	**Attr**	**Default**	**Description**
31:0	RW-V	0x0	pm_data_counter_low_value: PM data counter low value Low order bits [31:0] for PM data counter[1:0].

7.2.119 xppmcl[0:1]

XP PM Compare Low Bits

The value of PMD is compared to the value of PMC. If PMD is greater than PMC, this status is reflected in the PERFCON register and/or on the GE[3:0] as selected in the Event Status Output field of the PMR register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x488, 0x48c**				

Bit	Attr	Default	Description
31:0	RW-V	0xffffffff	pm_compare_low_value: PM compare low value Low order bits [31:0] for PM compare register [1:0].

7.2.120 xppmdh

XP PM Data High Bits

This register contains the high nibbles from each of the PMD 36-bit counter register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x490**				

Bit	Attr	Default	Description
11:8	RW-V	0x0	high_nibble_pex_counter1_value: High Nibble PEX Counter1 value High order bits [35:32] of the 36-bit PM Data1 register.
3:0	RW-V	0x0	high_nibble_pex_counter0_value: High Nibble PEX Counter0 value High order bits [35:32] of the 36-bit PM Data0 register.

7.2.121 xppmch

XP PM Compare High Bits

This register contains the high nibbles from each of the PMC 36-bit compare registers.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x492**				

Bit	Attr	Default	Description
11:8	RW-V	0xf	high_nibble_pex_compare1_value: High Nibble PEX Compare1 value High order bits [35:32] of the 36-bit PM Compare1 register.
3:0	RW-V	0xf	high_nibble_pex_compare0_value: High Nibble PEX Compare0 value High order bits [35:32] of the 36-bit PM Compare0 register.

7.2.122 xppmr[0:1]

XP PM Response Control

The PMR register controls operation of its associated counter, and provides overflow or maximum compare status information.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x494, 0x498**				

Bit	Attr	Default	Description
31:31	RV	0x0	Reserved
30:30	RW	0x0	not_greater_than_comparison: Not greater than comparison 0: PMC will compare a greater than function. When clear the perfmon status will assert when the PMD is greater than the PMC. 1: PMC will compare with NOT (greater than) function. When set the perfmon status will assert when the PMD is less than or equal to the PMC.
29:29	RW	0x0	force_pmd_counter_to_add_zero_to_input: Force PMD counter to add zero to input This feature is used with the queue measurement bus. When this bit is set the value on the queue measurement bus is added to zero so the result in PMD will always reflect the value from the queue measurement bus. 0: Do not add zero. Normal PerfMon operation. 1: Add zero with input queue bus.
28:28	RW	0x0	latched_count_enable_select: Latched Count Enable Select 0: Normal PM operation. Use CENS as count enable. 1: Use Latched count enable from queue empty events
27:27	RW	0x0	reset_pulse_enable: Reset Pulse Enable Setting this bit will select a pulsed version of the reset signal source in the reset block. 0: Normal reset signaling 1: Select a pulsed reset from the reset signal sources.
26:24	RV	0x0	Reserved
23:22	RW	0x0	dfx_byte_lane_selection_for_perfmon:
21:21	RW	0x0	local_dft_event_select:

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x494, 0x498**				

Bit	Attr	Default	Description
20:19	RW	0x0	event_group_selection: Event Group Selection Selects which event register to use for performance monitoring.00: Bus events (XPMEVL,H register) and also Resource Utilizations (XP_PMER Registers) when all XP_PMEH and XP_PMEL Registers are set to '0'. that is, When monitoring PMER events, all PMEV events are to be deselected; when monitoring PMEV events, all PMER events are to be deselected. 01: Reserved 10: Queue measurement (in the XPPMER register). **Note**: To enable FIFO queue histogramming write bit field CNTMD ='11' and select queues in the XPPMER register. 11: Reserved
18:17	RW	0x0	count_event_select: Count Event Select Selects the condition for incrementing the performance monitor counter. 00: Event source selected by PMEV{L,H} 01: Partner event status (maximum compare or overflow) 10: All clocks when enabled 11: Reserved
16:16	RW	0x0	event_polarity_invert: Event Polarity Invert This bit inverts the polarity of the conditioned event signal. 0: No inversion 1: Invert the polarity of the conditioned event signal
15:14	RW	0x0	count_mode: Count Mode This field sets how the events will be counted. 00: Count clocks when event is logic high. Counting is level sensitive, whenever the event is logic 1 the counter is enabled to count. 01: Count rising edge events. Active low signals should be inverted with EVPOLINV for correct measurements. 10: Latch event and count clocks continuously. After the event is asserted, latch this state and count clocks continuously. The latched state of this condition is cleared by xxxPMRx.CNTRST bit, or PERFCON.GBRST, or GE[3:0]. 11: Enable FIFO (push/pop) queue histogram measurement. This mode will enable histogram measurements on PM0. This mode enable logic to perform the function listed in the table below. The measurement cycle will not begin until the Qempty signal is asserted. Refer to xref. FIFO queue histogram table FIFOn_Push.......FIFOn_POP...........PMD Adder control ....0..........................0.......................Add zero ....1..........................0.......................Add queue bus value* ....0..........................1.......................Sub queue bus value* ....1..........................1.......................Add zero The latched condition of the Qempty signal cannot be cleared by PMR.CLREVLAT. A new measurement cycle requires clearing all counters and the latched value by asserting either PMRx.CNTRST or PERFCON.GBRST.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x494, 0x498**				

Bit	Attr	Default	Description
13:11	RW	0x0	counter_enable_source: Counter enable source These bits identify which input enables the counter. Default value disables counting. 000: Disabled 001: Local Count Enabled (LCEN). This bit is always a logic 1. 010: Partner counter’s event status (maximum compare or overflow) 011: Reserved 100: GE[0], from the Global Debug Event Block 101: GE[1], from the Global Debug Event Block 110: GE[2], from the Global Debug Event Block 111: GE[3], from the Global Debug Event Block **Note**: Address/Header MatchOut signal must align with PMEVL,H events for this to be effective.
10:8	RW	0x0	reset_event_select: Reset Event Select Counter and event status will reset and counting will continue. 000: No reset condition 001: Partner’s event status: When the partner counter causes an event status condition to be activated, either by a counter overflow or maximum comparison, then this counter will reset and continue counting. 010: Partners PME register event: When the partner counter detects a match condition which meets its selected PME register qualifications, then this counter will reset and continue counting. 011: This PM counter’s status output. 100: GE[0], from the Global Debug Event Block. 101: GE[1], from the Global Debug Event Block. 110: GE[2], from the Global Debug Event Block. 111: GE[3], from the Global Debug Event Block.
7:6	RW	0x0	compare_mode: Compare Mode This field defines how the PMC (compare) register is to be used. 00: compare mode disabled (PMC register not used) 01: maximum compare only: The PMC register value is compared with the counter value. If the counter value is greater then the Compare Status (CMPSTAT) will be set. 10: maximum compare with update of PMC at end of sample: The PMC register value is compared with the counter value, and if the counter value is greater, the PMC register is updated with the counter value. **Note**: The Compare Status field is not affected in this mode. 11: Reserved
5:5	RW	0x0	pm_status_signal_output: PM Status Signal Output 0: Level output from status/overflow signals. 1: Pulsed output from status/overflow signals.
4:3	RW	0x0	cto: PerfMon Trigger Output This field selects what the signal is communicated to the chip’s event logic structure. 00: No cluster trigger output from PerfMons or header match. 01: PM Status. 10: PM Event Detection. 11: Reserved

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x494, 0x498**				

Bit	Attr	Default	Description
2:2	RW1C	0x0	compare_status: Compare Status This status bit captures a count compare event. The Compare Status field can be programmed to allow this bit to be driven to Global Event (GE[3:0]) signals which will then distribute the event to the debug logic. 0: no event 1: count compare - PMD counter greater than PMC register when in compare mode. This bit remains set once an event is reported even though the original condition is no longer valid. Writing a logic '1' clears the bit.
1:1	RW1C	0x0	overflow_status_bit: Overflow Status Bit This status bit captures the overflow event from the PMD counter.This bit remains set once an event is reported even though the original condition is no longer valid. Writing a logic '1' clears the bit.
0:0	RW	0x0	counter_reset: Counter Reset Setting this bit resets the PMD counter, the associated adder storage register and the count mode state latch (see bits CNTMD) to the default state. It does not change the state of this PMR register, the event selections, or the value in the compare register. **Note**: This bit must be cleared by software, otherwise the counters remain in reset. There is also a reset bit in the PERFCON register which clears all PM registers including the PMR.

7.2.123 xppmevl[0:1]

XP PM Events Low

Selections in this register correspond to fields within the PCIe header. Each field selection is logically combined according to the match equation. The qualifications for fields in this register are listed below.

Note: The bit selections are generic for packet and for either inbound or outbound direction.

Because of this, there will be bit fields that do not make sense. For these packet matching situations the user should select "Either" which acts as a don't care for the match equation

PCIe PerfMon Match Equation

PMEV Match = ((IO_Cfg_Write_event + IO_Cfg_Read_event _+ Mem_Write_event + Mem_Read_event + Trusted_write_event + Trusted_read_event + General_event) and INOUTBND) + GESEL

IO_Cfg_Write_event = (REQCMP[0] and CMPR[1] and RDWR[1] and DATALEN and (TTYP[2] + (TTYP[1] and CFGTYP)))

IO_Cfg_Read_event = (REQCMP[0] and CMPR[1] and RDWR[0] and DATALEN and (TTYP[2] + (FMTTYP[1] and CFGTYP)))

Mem_Write_event = (REQCMP[0] and CMPR[0] and RDWR[1] and DATALEN and TTYP[3] and LOCK and EXTADDR and SNATTR)

Note: An outbound memory write does not have a snoop attribute as an inbound memory write has. So the user should set SNATTR="11" for outbound memory write transaction event counting.

Mem_Read_event = (REQCMP[0] and CMPR[1] and RDWR[0] and DATALEN and ((TTYP[3] and LOCK and EXTADDR and SNATTR) + TTYP[2] + (TTYP[1] and CFGTYP)))

Note: For outbound memory reads there is no concept of issuing a snoop cycle. The user should select SNATTR="11" for either snoop attribute.

Msg_event = (TTYP[0] and DND)

(INOUTBND[0] and (MatchEq) + (IOBND[1] and (MatchEq)

Setting both bits in INOUTBND is acceptable however the performance data gathered will not be accurate since once one header can be counted at a time.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x49c, 0x4a0**				

Bit	Attr	Default	Description
31:30	RW	0x0	data_or_no_data_attribute: Data or no data attribute x1: Request/completion/message with data 1x: Request/completion/message packet without data
29:28	RW	0x0	snoop_attribute: Snoop Attribute x1: No snoop required1x: Snoop required 11: Either
27:26	RW	0x0	request_or_completion_packet_selection: Request or Completion Packet Selection x1: Request packet 1x: Completion packet 11: Either
25:24	RW	0x0	read_or_write_selection: Read or Write Selection x1: Read 1x: Write 11: Either
23:22	RW	0x0	request_packet_only: Completion Required x1: No completion required 1x: Completion required 11: Either
21:20	RW	0x0	lock_attribute_selection: Lock Attribute Selection x1: No lock 1x: Lock 11: Either

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x49c, 0x4a0**				

Bit	Attr	Default	Description
19:18	RW	0x0	extended_addressing_header: Extended Addressing Header x1: 32b addressing 1x: 64b addressing 11: Either
17:16	RW	0x0	cfgtyp: Configuration Type x1: Type 0 1x: Type 1 11: Either
15:11	RW	0x0	fmttyp: Transaction Type Encoding 1_xxxx: Trusted x_1xxx: Memory x_x1xx: I/O x_xx1x: Configuration x_xxx1: Messages 1_1111: Any transaction type
10:4	RW	0x0	data_length: Data Length 1xx_xxxx: (129 to 256 bytes) x1x_xxxx: (65 to 128 bytes) xx1_xxxx: (33 to 64 bytes) xxx_1xxx: (17 to 32 bytes) xxx_x1xx: (9 to 16 bytes) xxx_xx1x: (0 to 8 bytes) xxx_xxx1: 0 bytes, used for a special zero length encoded packets 111_1111: Any Data length
3:0	RW	0x0	for_completion_packet_or_message_encoding_for_request_packet: Completion Status. 1xxx: Completer abort x1xx: Configuration request retry status (only used for inbound completions) xx1x: Unsupported request xxx1: Successful completion 1111: Any status The completion feature is not supported . This field should not be used by software (reserved): write 0 always, read return random.

7.2.124 xppmevh[0:1]

XP PM Events High

Selections in this register correspond to fields within the PEX packet header. Each field selection is ANDed with all other fields in this register including the XPPMEVL except for the Global Event signals. These signals are OR'ed with any event in the XPPMEVL and enables for debug operations requiring the accumulation of specific debug signals.The qualifications for fields in this register are as follows:

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x4a4, 0x4a8**				

Bit	Attr		Default
31:8	RV	0x0	Reserved
7:2	RW	0x0	global_event_selection: Global Event Selection Selects which GE[3:0] is used for event counting. This field is OR'd with other fields in this register. The GEs cannot be qualified with other PerfMon signals.If more than 1 GE is selected then the resultant event is the OR between each GE. However, properly counting Global Event based on design, XP PM Response Control Register bit [13:11] CENS must be set to choose GE[3:0] and also bit[18:17] CNTEVSEL must be set to 2'b10. 1x_xxxx: GE[5] x1_xxxx: GE[4] xx_1xxx: GE[3] xx_x1xx: GE[2] xx_xx1x: GE[1] xx_xxx1: GE[0]
1:0	RW	0x0	inbound_or_outbound_selection Inbound or Outbound Selection Selects which path to count transactions.1x: Outbound x1: Inbound (from PCI bus) 11: Either

7.2.125 xppmer[0:1]

XP PM Resource Event.

This register is used to select queuing structures for measurement. Use of this event register is mutually exclusive with the XPPMEV{L,H} registers. The Event Register Select field in the PMR register must select this register for to enable monitoring operations of the queues.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Bus:	**0**	**Device:**	**2**	**Function:**	**0**
Bus:	**0**	**Device:**	**3**	**Function:**	**0**
Offset:	**0x4ac, 0x4b0**				

Bit	Attr	Default	Description
20:17	RW	0x0	xp_resource_assignment: This selects which PCI Express* links are being monitored.A logic 1 selects that PCIe link for monitoring. 1000: Select NA / PXP6 / PXP10 (depending on device number) for monitoring. 0100: Select PXP2 / PXP5 / PXP9 (depending on device number) for monitoring. 0010: Select PXP1 / PXP4 / PXP8 (depending on device number) for monitoring. 0001: Select PXP / PXP3 / PXP7 (depending on device number) for monitoring.
16:13	RW	0x0	link_send_utilization: This level signal that is active when the link could send a packet or an idle. The choices are a logic idle flit, a link layer packet, or a transaction layer packet. The user can count the number of clocks that the link is not active by inverting this signal in the event conditioning logic (PMR.EVPOLINV = 1).The selection listed combines all the links for clarity. If the user is operating on XP3 then the bit field selects Links[6:3] only. 0000: No event selected 1000: Link 6 (xp3), link 10 (xp7), reserved, reserved 0100: Link 5 (xp3), link 9 (xp7), reserved, reserved 0010: Link 4 (xp3), link 8 (xp7), port 2 (xp0), reserved 0001: Link 3 (xp3), link 7 (xp7), link 1 (xp0), link 0 (xp0 -DMI)
7:6	RW	0x0	flowcntrclass:
5:0	RW	0x0	qbussel: Queue Measurement Bus Select: This field selects a queue to monitor. These queues are connected the QueueMeasBus that is derived from the difference in the write and read pointers. 000000: No queues selected --- 010001: xp0, xp3, xp7 - Inbound data payload 010010: xp1, xp4, xp8 - Inbound data payload 010100: xp2, xp5, xp9 - Inbound data payload 011000: N/A, xp6, xp10 - Inbound data payload 100001: xp0, xp3, xp7 - Outbound data payload 100010: xp1, xp4, xp8 - Outbound data payload 100100: xp2, xp5, xp9 - Outbound data payload 101000: N/A, xp6, xp10 - Outbound data payload others: reserved N/A: not applicable.

7.3 Device 0 Function 0 Region DMIRCBAR

DMI Root Complex Registers Block (RCRB). This block is mapped into memory space, using register DMIRCBAR [Device 0:Function 0, offset 0x50].

Table 7-6. Integrated I/O Device 0 Function 0 Region DMIRCBAR Register Address Map

Register Name	Offset	Size
dmivc0rcap	0x10	32
dmivc0rctl	0x14	32
dmivc0rsts	0x1a	16
dmivc1rcap	0x1c	32
dmivc1rctl	0x20	32
dmivc1rsts	0x26	16
dmivcprcap	0x28	32
dmivcprctl	0x2c	32
dmivcprsts	0x32	16
dmivcmrcap	0x34	32
dmivcmrctl	0x38	32
dmivcmrsts	0x3e	16
dmivc1cdtthrottle	0x60	32
dmivcpcdtthrottle	0x64	32
dmivcmcdtthrottle	0x68	32

7.3.1 dmivc0rcap

DMI VC0 Resource Capability

Type: MEM **Bus:** 0 **Offset:** 0x10		**PortID:** 8’h7e **Device:** 0	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:16	RO	0x0	maxtimeslots: Maximum Time Slots
15:15	RO	0x0	rejsnpt: Reject Snoop Transactions 0: Transactions with or without the No Snoop bit set within the TLP header are allowed on this VC. 1: Any transaction without the No Snoop bit set within the TLP header will be rejected as an Unsupported Request.
14:0	RV	0x0	Reserved

7.3.2 dmivc0rctl

DMI VC0 Resource Control

Controls the resources associated with PCI Express* Virtual Channel 0.

Type: **MEM** **Bus:** **0** **Offset:** **0x14**		**PortID:** **8'h7e** **Device:** **0**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
31:31	RO	0x1	vc0e: Virtual Channel 0 Enable For VC0 this is hardwired to 1 and read only as VC0 can never be disabled.
30:27	RV	0x0	Reserved
26:24	RO	0x0	vc0id: Virtual Channel 0 ID Assigns a VC ID to the VC resource. For VC0 this is hardwired to 0 and read only.
23:8	RV	0x0	Reserved
7:7	RO	0x0	tc7vc0m: Traffic Class 7/ Virtual Channel 0 Map Traffic Class 7 is always routed to VCm.
6:1	RW-LB	0x3f	tcvc0m: Traffic Class/Virtual Channel 0 Map Indicates the TCs (Traffic Classes) that are mapped to the VC resource. Bit locations within this field correspond to TC values.For example, when bit 6 is set in this field, TC6 is mapped to this VC resource. When more than one bit in this field is set, it indicates that multiple TCs are mapped to the VC resource. In order to remove one or more TCs from the TC/VC Map of an enabled VC, software must ensure that no new or outstanding transactions with the TC labels are targeted at the given Link.
0:0	RO	0x1	tc0vc0m: Traffic Class 0/Virtual Channel 0 Map Traffic Class 0 is always routed to VC0.

7.3.3 dmivc0rsts

DMI VC0 Resource Status

Reports the Virtual Channel specific status.

Type: **MEM** **Bus:** **0** **Offset:** **0x1a**		**PortID:** **8'h7e** **Device:** **0**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
15:2	RV	0x0	Reserved

Type: MEM Bus: 0 Offset: 0x1a		PortID: 8’h7e Device: 0	Function: 0
Bit	**Attr**	**Default**	**Description**
1:1	RO-V	0x1	vc0np: Virtual Channel 0 Negotiation Pending 0: The VC negotiation is complete. 1: The VC resource is still in the process of negotiation (initialization or disabling). This bit indicates the status of the process of Flow Control initialization. It is set by default on Reset, as well as whenever the corresponding Virtual Channel is Disabled or the Link is in the DL_Down state. It is cleared when the link successfully exits the FC_INIT2 state. **BIOS Requirement:** Before using a Virtual Channel, software must check whether the VC Negotiation Pending fields for that Virtual Channel are cleared in both Components on a Link.
0:0	RV	0x0	Reserved

7.3.4 dmivc1rcap

DMI VC1 Resource Capability

Type: MEM Bus: 0 Offset: 0x1c		PortID: 8’h7e Device: 0	Function: 0
Bit	**Attr**	**Default**	**Description**
15:15	RO	0x1	rejsnpt: Reject Snoop Transactions 0: Transactions with or without the No Snoop bit set within the TLP header are allowed on this VC. 1: Any transaction without the No Snoop bit set within the TLP header will be rejected as an Unsupported Request.
14:0	RV	0x0	Reserved

7.3.5 dmivc1rctl

DMI VC1 Resource Control

Controls the resources associated with PCI Express* Virtual Channel 1.

Type: MEM **Bus:** 0 **Offset:** 0x20		**PortID:** 8'h7e **Device:** 0	**Function:** 0

Bit	Attr	Default	Description
31:31	RW-LB	0x0	vc1e: Virtual Channel 1 Enable 0: Virtual Channel is disabled. 1: Virtual Channel is enabled. See exceptions below. Software must use the VC Negotiation Pending bit to check whether the VC negotiation is complete. When VC Negotiation Pending bit is cleared, a 1 read from this VC Enable bit indicates that the VC is enabled (Flow Control Initialization is completed for the PCI Express* port). A 0 read from this bit indicates that the Virtual Channel is currently disabled. **BIOS Requirement:** 1. To enable a Virtual Channel, the VC Enable bits for that Virtual Channel must be set in both Components on a Link. 2. To disable a Virtual Channel, the VC Enable bits for that Virtual Channel must be cleared in both Components on a Link. 3. Software must ensure that no traffic is using a Virtual Channel at the time it is disabled. 4. Software must fully disable a Virtual Channel in both Components on a Link before re-enabling the Virtual Channel.
30:27	RV	0x0	Reserved
26:24	RW-LB	0x1	vc1id: Virtual Channel 1 ID Assigns a VC ID to the VC resource. Assigned value must be non-zero. This field cannot be modified when the VC is already enabled.
23:8	RV	0x0	Reserved
7:7	RO	0x0	tc7vc1m: Traffic Class 7/ Virtual Channel 1 Map Traffic Class 7 is always routed to VCm.
6:1	RW-LB	0x0	tcvc1m: Traffic Class/Virtual Channel 1 Map Indicates the TCs (Traffic Classes) that are mapped to the VC resource. Bit locations within this field correspond to TC values.For example, when bit 6 is set in this field, TC6 is mapped to this VC resource. When more than one bit in this field is set, it indicates that multiple TCs are mapped to the VC resource. In order to remove one or more TCs from the TC/VC Map of an enabled VC, software must ensure that no new or outstanding transactions with the TC labels are targeted at the given Link.
0:0	RO	0x0	tc0vc1m: Traffic Class 0/Virtual Channel 0 Map Traffic Class 0 is always routed to VC0.

7.3.6 dmivc1rsts

DMI VC1 Resource Status

Reports the Virtual Channel specific status.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x26**				

Bit	Attr	Default	Description
15:2	RV	0x0	Reserved
1:1	RO-V	0x1	vc1np: Virtual Channel 1 Negotiation Pending 0: The VC negotiation is complete.1: The VC resource is still in the process of negotiation (initialization or disabling). This bit indicates the status of the process of Flow Control initialization. It is set by default on Reset, as well as whenever the corresponding Virtual Channel is Disabled or the Link is in the DL_Down state. It is cleared when the link successfully exits the FC_INIT2 state. **BIOS Requirement:** Before using a Virtual Channel, software must check whether the VC Negotiation Pending fields for that Virtual Channel are cleared in both Components on a Link.
0:0	RV	0x0	Reserved

7.3.7 dmivcprcap

DMI VCP Resource Capability

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x1a**				

Bit	Attr	Default	Description
15:15	RO	0x0	rejsnpt: Reject Snoop Transactions 0: Transactions with or without the No Snoop bit set within the TLP header are allowed on this VC.1: Any transaction without the No Snoop bit set within the TLP header will be rejected as an Unsupported Request.
14:0	RV	0x0	Reserved

7.3.8 dmivcprctl

DMI VCP Resource Control

Controls the resources associated with the DMI Private Channel (VCp).

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x1a**				

Bit	Attr	Default	Description
31:31	RW-LB	0x0	vcpe: Virtual Channel Private Enable 0: Virtual Channel is disabled. 1: Virtual Channel is enabled. See exceptions below. Software must use the VC Negotiation Pending bit to check whether the VC negotiation is complete. When VC Negotiation Pending bit is cleared, a 1 read from this VC Enable bit indicates that the VC is enabled (Flow Control Initialization is completed for the PCI Express* port). A 0 read from this bit indicates that the Virtual Channel is currently disabled. **BIOS Requirement:** 1. To enable a Virtual Channel, the VC Enable bits for that Virtual Channel must be set in both Components on a Link. 2. To disable a Virtual Channel, the VC Enable bits for that Virtual Channel must be cleared in both Components on a Link. 3. Software must ensure that no traffic is using a Virtual Channel at the time it is disabled. 4. Software must fully disable a Virtual Channel in both Components on a Link before re-enabling the Virtual Channel.
30:27	RV	0x0	Reserved
26:24	RW-LB	0x2	vcpid: Virtual Channel Private ID Assigns a VC ID to the VC resource. This field cannot be modified when the VC is already enabled. No private VCs are precluded by hardware and private VC handling is implemented the same way as non-private VC handling.
23:8	RV	0x0	Reserved
7:7	RO	0x0	tc7vcpm: Traffic Class 7/ Virtual Channel 0 Map Traffic Class 7 is always routed to VCm.
6:1	RW-LB	0x0	tcvcpm: Traffic Class/Virtual Channel private Map Indicates the TCs (Traffic Classes) that are mapped to the VC resource. Bit locations within this field correspond to TC values.For example, when bit 6 is set in this field, TC6 is mapped to this VC resource. When more than one bit in this field is set, it indicates that multiple TCs are mapped to the VC resource. In order to remove one or more TCs from the TC/VC Map of an enabled VC, software must ensure that no new or outstanding transactions with the TC labels are targeted at the given Link.
0:0	RO	0x0	tc0vcpm: Traffic Class 0/Virtual Channel Private Map Traffic Class 0 is always routed to VC0.

7.3.9 dmivcprsts

DMI VCP Resource Status

Reports the Virtual Channel specific status.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x32**				

Bit	Attr	Default	Description
15:2	RV	0x0	Reserved
1:1	RO-V	0x1	vcpnp: Virtual Channel Private Negotiation Pending 0: The VC negotiation is complete. 1: The VC resource is still in the process of negotiation (initialization or disabling). This bit indicates the status of the process of Flow Control initialization. It is set by default on Reset, as well as whenever the corresponding Virtual Channel is Disabled or the Link is in the DL_Down state. It is cleared when the link successfully exits the FC_INIT2 state. **BIOS Requirement:** Before using a Virtual Channel, software must check whether the VC Negotiation Pending fields for that Virtual Channel are cleared in both Components on a Link.
0:0	RV	0x0	Reserved

7.3.10 dmivcmrcap

DMI VCM Resource Capability

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x34**				

Bit	Attr	Default	Description
31:16	RV	0x0	Reserved
15:15	RO	0x1	rejsnpt: Reject Snoop Transactions 0: Transactions with or without the No Snoop bit set within the TLP header are allowed on this VC. 1: Any transaction without the No Snoop bit set within the TLP header will be rejected as an Unsupported Request.
14:0	RV	0x0	Reserved

7.3.11 dmivcmrctl

DMI VCM Resource Control

Controls the resources associated with PCI Express* Virtual Channel 0.

Type: **MEM** **Bus:** **0** **Offset:** **0x38**	**PortID:** **8’h7e** **Device:** **0**	**Function:** **0**	

Bit	Attr	Default	Description
31:31	RW-LB	0x0	vcme: Virtual Channel M Enable 0: Virtual Channel is disabled. 1: Virtual Channel is enabled. See exceptions below. Software must use the VC Negotiation Pending bit to check whether the VC negotiation is complete. When VC Negotiation Pending bit is cleared, a 1 read from this VC Enable bit indicates that the VC is enabled (Flow Control Initialization is completed for the PCI Express* port). A 0 read from this bit indicates that the Virtual Channel is currently disabled. **BIOS Requirement:** 1. To enable a Virtual Channel, the VC Enable bits for that Virtual Channel must be set in both Components on a Link. 2. To disable a Virtual Channel, the VC Enable bits for that Virtual Channel must be cleared in both Components on a Link. 3. Software must ensure that no traffic is using a Virtual Channel at the time it is disabled. 4. Software must fully disable a Virtual Channel in both Components on a Link before re-enabling the Virtual Channel.
30:27	RV	0x0	Reserved
26:24	RW-LB	0x0	vcmid: VCm ID
23:8	RV	0x0	Reserved
7:7	RO	0x1	tc7vcpm: Traffic Class 7/ Virtual Channel 0 Map Traffic Class 7 is always routed to VCm.
6:1	RO	0x0	tcvcmm: Traffic Class/Virtual Channel M Map No other traffic class is mapped to VCM
0:0	RO	0x0	tc0vcmm: Traffic Class 0 Virtual Channel Map

7.3.12 dmivimrsts

DMI VCM Resource Status

Reports the Virtual Channel specific status.

Type: **MEM** **Bus:** **0** **Offset:** **0x3e**	**PortID:** **8’h7e** **Device:** **0**	**Function:** **0**	

Bit	Attr	Default	Description
15:2	RV	0x0	Reserved

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x3e**				

Bit	Attr	Default	Description
1:1	RO-V	1b	vcmnp: Virtual Channel M Negotiation Pending 0: The VC negotiation is complete.1: The VC resource is still in the process of negotiation (initialization or disabling). This bit indicates the status of the process of Flow Control initialization. It is set by default on Reset, as well as whenever the corresponding Virtual Channel is Disabled or the Link is in the DL_Down state. It is cleared when the link successfully exits the FC_INIT2 state. **BIOS Requirement:** Before using a Virtual Channel, software must check whether the VC Negotiation Pending fields for that Virtual Channel are cleared in both Components on a Link.
0:0	RV	0x0	Reserved

7.3.13 dmivc1cdtthrottle

DMI VC1 Credit Throttle

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x60**				

Bit	Attr	Default	Description
31:24	RWS	0x0	prd: Posted Request Data VC1 Credit Withhold Number of VC1 Posted Data credits to withhold from being reported or used.
23:22	RV	0x0	Reserved
21:16	RWS	0x0	prh: Posted Request Header VC1 Credit Withhold Number of VC1 Posted Request credits to withhold from being reported or used.
15:8	RWS	0x0	nprd: Non-Posted Request Data VC1 Credit Withhold Number of VC1 Non-Posted Data credits to withhold from being reported or used.
7:6	RV	0x0	Reserved
5:0	RWS	0x0	nprh: Non-Posted Request Header VC1 Credit Withhold Number of VC1 Non-Posted Request credits to withhold from being reported or used.

7.3.14 dmivcpcdtthrottle

DMI VCp Credit Throttle

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x64**				

Bit	Attr	Default	Description
31:24	RWS	0x0	prd: Posted Request Data VCp Credit Withhold Number of VCp Posted Data credits to withhold from being reported or used.
23:22	RV	0x0	Reserved
21:16	RWS	0x0	prh: Posted Request Header VCp Credit Withhold Number of VCp Posted Request credits to withhold from being reported or used.
15:8	RWS	0x0	nprd: Non-Posted Request Data VCp Credit Withhold Number of VCp Non-Posted Data credits to withhold from being reported or used.
7:6	RV	0x0	Reserved
5:0	RWS	0x0	nprh: Non-Posted Request Header VCp Credit Withhold Number of VCp Non-Posted Request credits to withhold from being reported or used.

7.3.15 dmivcmcdtthrottle

DMI VCm Credit Throttle

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**0**	**Function:**	**0**
Offset:	**0x68**				

Bit	Attr	Default	Description
31:24	RWS	0x0	prd: Posted Request Data VCm Credit Withhold Number of VCm Posted Data credits to withhold from being reported or used.
23:22	RV	0x0	Reserved
21:16	RWS	0x0	prh: Posted Request Header VCm Credit Withhold Number of VCm Posted Request credits to withhold from being reported or used.
15:8	RWS	0x0	nprd: Non-Posted Request Data VCm Credit Withhold Number of VCm Non-Posted Data credits to withhold from being reported or used.
7:6	RV	0x0	Reserved
5:0	RWS	0x0	nprh: Non-Posted Request Header VCm Credit Withhold Number of VCm Non-Posted Request credits to withhold from being reported or used.

7.4 Device 5 Function 0

Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d), Address Mapping, System Management, Coherent Interface, Misc Registers.

Table 7-7. Integrated I/O Device 5 Function 0 Register Address Map (Sheet 1 of 2)

Register Name	Offset	Size
vid	0x0	16
did	0x2	16
pcicmd	0x4	16
pcists	0x6	16
rid	0x8	8
ccr	0x9	24
clsr	0xc	8
hdr	0xe	8
svid	0x2c	16
sdid	0x2e	16
capptr	0x34	8
intl	0x3c	8
intpin	0x3d	8
pxpcapid	0x40	8
pxpnxtptr	0x41	8
pxpcap	0x42	16
hdrtypectrl	0x80	8
mmcfg_base	0x84	32
mmcfg_limit	0x88	32
tseg	0xa8	64
genprotrange1_base	0xb0	64
genprotrange1_limit	0xb8	64
genprotrange2_base	0xc0	64
genprotrange2_limit	0xc8	64
tolm	0xd0	32
tohm	0xd4	64
ncmem_base	0xe0	64
ncmem_limit	0xe8	64
mencmem_base	0xf0	64
mencmem_limit	0xf8	64
cpubusno	0x108	32
lmmiol_base	0x10c	16
lmmiol_limit	0x10e	16
lmmioh_base	0x110	64
lmmioh_limit	0x118	64
genprotrange0_base	0x120	64

Table 7-7. Integrated I/O Device 5 Function 0 Register Address Map (Sheet 2 of 2)

Register Name	Offset	Size
genprotrange0_limit	0x128	64
cipctrl	0x140	32
cipsts	0x144	32
cipdcasad	0x148	32
cipintrc	0x14c	64
cipintrs	0x154	32
vtbar	0x180	32
vtgenctrl	0x184	16
vtisochctrl	0x188	32
vtgenctrl2	0x18c	32
iotlbpartition	0x194	32
vtuncerrsts	0x1a8	32
vtuncerrmsk	0x1ac	32
vtuncerrsev	0x1b0	32
vtuncerrptr	0x1b4	8
iiomiscctrl	0x1c0	64
ltdpr	0x290	32
lcfgbus_base	0x41c	8
lcfgbus_limit	0x41d	8
csipintrs	0x450	32

7.4.1 vid

Type: CFG **Bus:** 0 **Offset:** 0x0		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
15:0	RO	0x8086	vendor_identification_number: The value is assigned by PCI-SIG to Intel.

7.4.2 did

Type: CFG **Bus:** 0 **Offset:** 0x2		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
15:0	RO	0xe28	device_identification_number: Device ID values vary from function to function. Bits 15:8 are equal to 0x0E.

7.4.3 pcicmd

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	0
Offset:	0x4				

Bit	Attr	Default	Description
10:10	RO	0x0	intx_disable: N/A for these devices
9:9	RO	0x0	fast_back_to_back_enable: Not applicable to PCI Express* and is hardwired to 0
8:8	RO	0x0	serr_enable: This bit has no impact on error reporting from these devices
7:7	RO	0x0	idsel_stepping_wait_cycle_control: Not applicable to internal devices. Hardwired to 0.
6:6	RO	0x0	parity_error_response: This bit has no impact on error reporting from these devices
5:5	RO	0x0	vga_palette_snoop_enable: Not applicable to internal devices. Hardwired to 0.
4:4	RO	0x0	memory_write_and_invalidate_enable: Not applicable to internal devices. Hardwired to 0.
3:3	RO	0x0	special_cycle_enable: Not applicable. Hardwired to 0.
2:2	RO	0x0	bus_master_enable: Hardwired to 0 since these devices don't generate any transactions
1:1	RO	0x0	memory_space_enable: Hardwired to 0 since these devices don't decode any memory BARs
0:0	RO	0x0	io_space_enable: Hardwired to 0 since these devices don't decode any I/O BARs

7.4.4 pcists

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	0
Offset:	0x6				

Bit	Attr	Default	Description
15:15	RO	0x0	detected_parity_error: This bit is set when the device receives a packet on the primary side with an uncorrectable data error including a packet with poison bit set or an uncorrectable addresscontrol parity error. The setting of this bit is regardless of the Parity Error Response bit PERRE in the PCICMD register. R2PCIe will never set this bit.
14:14	RO	0x0	signaled_system_error: Hardwired to 0
13:13	RO	0x0	received_master_abort: Hardwired to 0
12:12	RO	0x0	received_target_abort: Hardwired to 0
11:11	RO	0x0	signaled_target_abort: Hardwired to 0

Type: **CFG** **Bus:** **0** **Offset:** **0x6**	**PortID:** **N/A** **Device:** **5**		**Function:** **0**
Bit	**Attr**	**Default**	**Description**
10:9	RO	0x0	devsel_timing: Not applicable to PCI Express*. Hardwired to 0.
8:8	RO	0x0	master_data_parity_error: Hardwired to 0
7:7	RO	0x0	fast_back_to_back: Not applicable to PCI Express*. Hardwired to 0.
5:5	RO	0x0	pci66mhz_capable: Not applicable to PCI Express*. Hardwired to 0.
4:4	RO	0x1	capabilities_list: This bit indicates the presence of a capabilities list structure
3:3	RO	0x0	intx_status: Hardwired to 0

7.4.5 rid

Type: **CFG** **Bus:** **0** **Offset:** **0x8**	**PortID:** **N/A** **Device:** **5**		**Function:** **0**
Bit	**Attr**	**Default**	**Description**
7:0	RO-V	0x0	revision_id: Reflects the Uncore Revision ID after reset. Reflects the Compatibility Revision ID after the BIOS writes 0x69 to any RID register in any processor function. **Implementation Note:** Read and write requests from the host to any RID register in any processor function are re-directed to the IIO cluster. Accesses to the CCR field are also redirected due to Dword alignment. It is possible that JTAG accesses are direct, so will not always be redirected.

7.4.6 ccr

Type: **CFG** **Bus:** **0** **Offset:** **0x9**	**PortID:** **N/A** **Device:** **5**		**Function:** **0**
Bit	**Attr**	**Default**	**Description**
23:16	RO-V	0x8	base_class: Generic Device
15:8	RO-V	0x80	sub_class: Generic Device
7:0	RO-V	0x0	register_level_programming_interface: Set to 00h for all non-APIC devices.

7.4.7 clsr

Type: CFG **Bus:** 0 **Offset:** 0xc		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
7:0	RW	0x0	cacheline_size: This register is set as RW for compatibility reasons only. Cacheline size is always 64B.

7.4.8 hdr

Type: CFG **Bus:** 0 **Offset:** 0xe		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
7:7	RO	0x1	multi_function_device: This bit defaults to 1b since all these devices are multi-function
6:0	RO	0x0	configuration_layout: This field identifies the format of the configuration header layout. It is Type 0 for all these devices. The default is 00h, indicating a 'endpoint device'.

7.4.9 svid

Type: CFG **Bus:** 0 **Offset:** 0x2c		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
15:0	RW-O	0x0	subsystem_vendor_identification_number: The default value specifies Intel but can be set to any value once after reset.

7.4.10 sdid

Type: CFG **Bus:** 0 **Offset:** 0x2e		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
15:0	RW-O	0x0	subsystem_device_identification_number: Assigned by the subsystem vendor to uniquely identify the subsystem

7.4.11 capptr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x34**				

Bit	Attr	Default	Description
7:0	RO	0x40	capability_pointer: Points to the first capability structure for the device which is the PCIe capability.

7.4.12 intl

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x3c**				

Bit	Attr	Default	Description
7:0	RO	0x0	interrupt_line: N/A for these devices

7.4.13 intpin

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x3d**				

Bit	Attr	Default	Description
7:0	RO	0x0	interrupt_pin: N/A since these devices do not generate any interrupt on their own

7.4.14 pxpcapid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x40**				

Bit	Attr	Default	Description
7:0	RO	0x10	capability_id: Provides the PCI Express* capability ID assigned by PCI-SIG.

7.4.15 pxpnxtptr

Type: CFG **Bus:** 0 **Offset:** 0x41			**PortID:** N/A **Device:** 5 **Function:** 0
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x0	next_ptr: This field is set to the PCI PM capability.

7.4.16 pxpcap

Type: CFG **Bus:** 0 **Offset:** 0x42			**PortID:** N/A **Device:** 5 **Function:** 0
Bit	**Attr**	**Default**	**Description**
13:9	RO	0x0	interrupt_message_number_n_a:
8:8	RO	0x0	slot_implemented_n_a:
7:4	RO	0x9	device_port_type: This field identifies the type of device. It is set to for the DMA to indicate root complex integrated endpoint device.
3:0	RO	0x2	capability_version: This field identifies the version of the PCI Express* capability structure. Set to 2h for PCI Express* and DMA devices for compliance with the extended base registers.

7.4.17 hdrtypectrl

PCI Header Type Control

Type: CFG **Bus:** 0 **Offset:** 0x80			**PortID:** N/A **Device:** 5 **Function:** 0
Bit	**Attr**	**Default**	**Description**
2:0	RW	0x0	clr_hdrmfd: When set, function#0 with in the indicated device shows a value of 0 for bit 7 of the HDR register, indicating a single function device. The BIOS sets this bit, when only function#0 is visible within the device, either because SKU reasons or the BIOS has hidden all functions but function#0 within the device by means of the DEVHIDE register. Bit 0 is for Device#1 Bit 1 is for Device#2 Bit 3 is for Device#3 Currently this is defined only for devices 1, 2 and 3 because in other devices it is expected that at least 2 functions are visible to OS or the entire device is hidden.

7.4.18 mmcfg_base

MMCFG Address Base

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x84**				

Bit	Attr	Default	Description
31:26	RW-LB	0x3f	mmcfg_base_addr: Indicates the base address which is aligned to a 64MB boundary.

7.4.19 mmcfg_limit

MMCFG Address Limit.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x88**				

Bit	Attr	Default	Description
31:26	RW-LB	0x0	mmcfg_limit_addr: Indicates the limit address which is aligned to a 64MB boundary. Any access that decodes to be between MMCFG.BASE<= Addr <= MMCFG.LIMIT targets the MMCFG region and is aborted by IIO. Setting the MMCFG.BASE greater than MMCFG.LIMIT, disables this region.

7.4.20 tseg

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xa8**				

Bit	Attr	Default	Description
63:52	RW-LB	0x0	limit: Indicates the limit address which is aligned to a 1MB boundary. Any access to falls within TSEG.BASE[31:20] <= Addr[31:20] <= TSEG.LIMIT[31:20] is considered to target the Tseg region and IIO aborts it. **Note:** Address bits 19:0 are ignored and not compared. The result is that BASE[19:0] is effectively 00000h and LIMIT is effectively FFFFFh. Setting the TSEG.BASE greater than the limit, disable this region.
31:20	RW-LB	0xfe0	base: Indicates the base address which is aligned to a 1MB boundary. Bits [31:20] corresponds to A[31:20] address bits.

7.4.21 genprotrange[1:0]_base

Generic Protected Memory Range X Base Address. (X = 1, 0)

Type: CFG **Bus:** 0 **Offset:** 0xb0, 0x120	**PortID:** N/A **Device:** 5		**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:51	RV	0x0	Reserved.
50:16	RW-LB	0x7ffffffff	base_address: [50:16] of generic memory address range that needs to be protected from inbound dma accesses. The protected memory range can be anywhere in the memory space addressable by the processor. Addresses that fall in this range that is, GenProtRange.Base[63:16] <= Address [63:16] <= GenProtRange.Limit [63:16], are completer aborted by IIO. Setting the Protected range base address greater than the limit address disables the protected memory region. **Note:** This range is orthogonal to Intel VT-d specification defined protected address range. Since this register provides for a generic range, it can be used to protect any system dram region or MMIO region from DMA accesses. But the expected usage for this range is to abort all PCIe accesses to the PCI-Segments region.
15:0	RV	0x0	Reserved.

7.4.22 genprotrange[1:0]_limit

Generic Protected Memory Range X Limit Address. (X = 1, 0)

Type: CFG **Bus:** 0 **Offset:** 0xb8, 0x128	**PortID:** N/A **Device:** 5		**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:51	RV	0x0	rsvd:
31:16	RW-LB	0x0	limit_address: [50:16] of generic memory address range that needs to be protected from inbound dma accesses. The protected memory range can be anywhere in the memory space addressable by the processor. Addresses that fall in this range that is, GenProtRange.Base[63:16] <= Address [63:16] <= GenProtRange.Limit [63:16], are completer aborted by IIO. Setting the Protected range base address greater than the limit address disables the protected memory region. **Note:** This range is orthogonal to Intel VT-d specification defined protected address range. This register is programmed once at boot time and does not change after that, including any quiesce flows. Since this register provides for a generic range, it can be used to protect any system dram region from DMA accesses. The expected usage for this range is to abort all PCIe accesses to the PCI-Segments region.
15:0	RV	0x0	rsvd:

7.4.23 genprotrange2_base

Generic Protected Memory Range 2 Base Address.

Type: CFG **Bus:** 0 **Offset:** 0xc0		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:51	RV	0x0	Reserved.
50:16	RW-LB	0x7ffffffff	base_address: [50:16] of generic memory address range that needs to be protected from inbound dma accesses. The protected memory range can be anywhere in the memory space addressable by the processor. Addresses that fall in this range that is, GenProtRange.Base[63:16] <= Address [63:16] <= GenProtRange.Limit [63:16], are completer aborted by IIO. Setting the Protected range base address greater than the limit address disables the protected memory region. **Note:** This range is orthogonal to Intel VT-d specification defined protected address range. This register is programmed once at boot time and does not change after that, including any quiesce flows. This region is expected to be used to protect against PAM region accesses inbound, but could also be used for other purposes, if needed.
15:0	RV	0x0	Reserved.

7.4.24 genprotrange2_limit

Generic Protected Memory Range 2 Limit Address.

Type: CFG **Bus:** 0 **Offset:** 0xc8		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:51	RV	0x0	Reserved:
31:16	RW-LB	0x0	limit_address: [50:16] of generic memory address range that needs to be protected from inbound dma accesses. The protected memory range can be anywhere in the memory space addressable by the processor. Addresses that fall in this range that is, GenProtRange.Base[63:16] <= Address [63:16] <= GenProtRange. Limit [63:16], are completer aborted by IIO. Setting the Protected range base address greater than the limit address disables the protected memory region. **Note:** This range is orthogonal to Intel VT-d specification defined protected address range. This register is programmed once at boot time and does not change after that, including any quiesce flows. This region is expected to be used to protect against PAM region accesses inbound, but could also be used for other purposes, if needed.
15:0	RV	0x0	Reserved:

7.4.25 tolm

Top of Low Memory

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xd0**				

Bit	Attr	Default	Description
31:26	RW-LB	0x0	addr: TOLM Address. Indicates the top of low dram memory which is aligned to a 64MB boundary. A 32-bit transaction that satisfies ’0 <= Address[31:26] <= TOLM[31:26]’ is a transaction towards main memory.
25:0	RV	0x0	Reserved.

7.4.26 tohm

Top of High Memory.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xd4**				

Bit	Attr	Default	Description
63:26	RW-LB	0x0	addr: TOHM Address. Indicates the limit of an aligned 64MB granular region that decodes > 4GB addresses towards system dram memory. A 64-bit transaction that satisfies ’4G <= A[63:26] <= TOHM[63:26]’ is a transaction towards main memory. This register is programmed once at boot time and does not change after that, including during quiesce flows.
25:0	RV	0x0	Reserved.

7.4.27 ncmem_base

Non-Coherent Memory Base Address.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xe0**				

Bit	Attr	Default	Description
63:26	RW-LB	0x3ffffffff	addr: Non Coherent memory base address. Address bits [63:26] of an inbound address if it satisfies ’NcMem.Base[63:26] <= A[63:26] <= NcMem. This means that IIO cannot ever use ’allocating’ write commands for accesses to this region, over IDI. This, in effect, means that DCA/TH writes cannot ever target this address region. The range indicated by the Non-coherent memory base and limit registers does not necessarily fall within the low dram or high dram memory regions as described by means of the corresponding base and limit registers. Usage Model for this range is ROL. But accesses to this range will use non-allocating reads and writes, when enabled. This register is programmed once at boot time and does not change after that, including any quiesce flows
25:0	RV	0x0	Reserved.

7.4.28 ncmem_limit

Non-Coherent Memory Limit.

Type: **CFG**	**PortID:** **N/A**		
Bus: **0**	**Device:** **5**	**Function:** **0**	
Offset: **0xe8**			
Bit	**Attr**	**Default**	**Description**
63:26	RW-LB	0x0	addr: Non Coherent memory limit address. Address bits [63:26] of an inbound address if it satisfies 'NcMem.Base[63:26] <= A[63:26] <= NcMem.This means that IIO cannot ever use 'allocating' write commands for accesses to this region, over IDI. This in effect means that DCA/TH writes cannot ever target this address region. The range indicated by the Non-coherent memory base and limit registers does not necessarily fall within the low dram or high dram memory regions as described by means of the corresponding base and limit registers. This register is programmed once at boot time and does not change after that, including any quiesce flows.
25:0	RV	0x0	Reserved.

7.4.29 mencmem_base

Intel® Management Engine (Intel® ME) Non-Coherent Memory Base Address.

Type: **CFG**	**PortID:** **N/A**		
Bus: **0**	**Device:** **5**	**Function:** **0**	
Offset: **0xf0**			
Bit	**Attr**	**Default**	**Description**
63:19	RW-LB	0x1fffffffffff	addr: Intel® Management Engine (Intel® ME) UMA Base Address. Indicates the base address which is aligned to a 1MB boundary. Bits [63:19] corresponds to A[63:19] address bits.
18:0	RV	0x0	Reserved.

7.4.30 mencmem_limit

Intel® Management Engine (Intel® ME) Non-Coherent Memory Base Limit.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xf8**				

Bit	Attr	Default	Description
63:19	RW-LB	0x0	addr: Intel ME UMA Limit Address. Indicates the limit address which is aligned to a 1MB boundary. Bits [63:19] corresponds to A[63:19] address bits.Any address that falls within MENCMEMBASE <= Addr <= MENCMEMLIMIT range is considered to target the UMA range. Setting the MCNCMEMBASE greater than the MCNCMEMLIMIT disables this range. The range indicated by this register must fall within the low dram or high dram memory regions as described by means of the corresponding base and limit registers.
18:0	RV	0x0	Reserved.

7.4.31 cpubusno

Processor Internal Bus Numbers.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x108**				

Bit	Attr	Default	Description
24:17	RW-LB	0x0	segment:
16:16	RW-LB	0x0	valid: 1: IIO claims PCI config accesses from ring if: the bus# matches the value in bits 7:0 of this register and Dev# >= 16 OR the bus# does not match either the value in bits 7:0 or 15:8 of this register 0: IIO does not claim PCI config accesses from ring
15:8	RW-LB	0x0	bus1: Is the internal bus# of rest of Uncore. All devices are claimed by UBOX on behalf of this component. Devices that do not exist within this component on this bus number are master aborted by the UBOX.
7:0	RW-LB	0x0	bus0: Is the internal bus# of IIO and also PCH. Configuration requests that target Devices 16-31 on this bus number must be forwarded to the PCH by the IIO. Devices 0-15 on this bus number are claimed by the UBOX to send to IIO internal registers. UBOX master aborts devices 8-15 automatically, since these devices do not exist.

7.4.32 lmmiol_base

Local MMIO Low Base.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x10c**				

Bit	Attr	Default	Description
15:8	RW-LB	0x0	base: Corresponds to A[31:24] of MMIOL base address. An inbound memory address that satisfies ‘local MMIOL base[15:8] <= A[31:24] <= local MMIOL limit[15:8]’ is treated as a local peer-to-peer transaction that do not cross coherent interface. **Note:** Setting LMMIOL.BASE greater than LMMIOL.LIMIT disables local MMIOL peer-to-peer. This register is programmed once at boot time and does not change after that, including any quiesce flows.

7.4.33 lmmiol_limit

Local MMIO Low Limit.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x10e**				

Bit	Attr	Default	Description
15:8	RW-LB	0x0	limit: Corresponds to A[31:24] of MMIOL limit. An inbound memory address that satisfies ‘local MMIOL base[15:8] <= A[31:24] <= local MMIOL limit[15:8]’ is treated as a local peer-to-peer transaction that does not cross the coherent interface. **Note:** Setting LMMIOL.BASE greater than LMMIOL.LIMIT disables local MMIOL peer-to-peer. This register is programmed once at boot time and does not change after that, including any quiesce flows.

7.4.34 lmmioh_base

Local MMIO High Base.

Type: **CFG** **Bus:** **0** **Offset:** **0x110**	**PortID:** **N/A** **Device:** **5**	**Function:** **0**	
Bit	**Attr**	**Default**	**Description**
50:26	RW-LB	0x0	base: Corresponds to A[50:26] of MMIOH base. An inbound memory address that satisfies local MMIOH base [50:26] <= A[63:26] <= local MMIOH limit [50:26] is treated as a local peer-to-peer transaction that does not cross the coherent interface. **Notes:** • Setting LMMIOH.BASE greater than LMMIOH.LIMIT disables local MMIOH peer-to-peer. • This register is programmed once at boot time and does not change after that, including any quiesce flows.

7.4.35 lmmioh_limit

Local MMIO High Limit.

Type: **CFG** **Bus:** **0** **Offset:** **0x118**	**PortID:** **N/A** **Device:** **5**	**Function:** **0**	
Bit	**Attr**	**Default**	**Description**
50:26	RW-LB	0x0	limit: Corresponds to A[50:26] of Local MMIOH Limit (and Base) Address. An inbound memory address that satisfies the Local MMIO Base Address [50:26] <=A[63:26] <= Local MMIOH Limit Address [50:26], with A[63:51] equal to zero, is treated as a local peer-to-peer transaction that does not cross the coherent interface (ring).

7.4.36 cipctrl

Coherent Interface Protocol Control.

Type: CFG Bus: 0 Offset: 0x140		PortID: N/A Device: 5	Function: 0
Bit	**Attr**	**Default**	**Description**
31:31	RW	0x0	flushpendwr: Whenever this bit is written to 1 (regardless what the current value of this bit is), IRP block first clears bit 0 in CIPSTS register and takes a snapshot of the currently pending write transactions to dram in Write Cache, wait for them to complete fully (that is, deallocate the corresponding Write CacheRRB entry) and then set bit 0 in CIPSTS register.
28:28	RW	0x0	diswrupdtflow: When set, PCIWriteUpdate command is never issued on IDI and the writes that triggered this flow would be treated as 'normal' writes and the rules corresponding to the 'normal writes' apply.
16:16	RW	0x0	rrbsize_3: This is the MSB bit for the rrbsize. The lower 3-bits of the rrbsize reside in CIPCTRL[11:9]
15:15	RW	0x0	rd_merge_enable:
14:12	RW	0x0	socketid: This is the BIOS programmed field that indicates the 'SocketID' of this particular socket. 'SocketID' is the unique value that each socket in the system gets for DCADIO target determination. Normally this value is the same as the APICID[7:5] of the cores in the socket, but it can be other values as well, if system topology were to not allow that straight mapping. IIO uses strapped NodeID to compare against the target NodeID determined by using the target SocketID value as a lookup into the CIPDCASAD register. If there is a match, then a PCIDCAHint is not sent (since the data is already located in the same LLC). This register is not used for this comparison. It is not used by hardware at all.
11:9	RW	0x0	rrbsize: Specifies the number of entries used in each half of the write cache. The default is to use all entries. 0000: 104 each side (208 total) 0001: 96 each side (192 total) 0010: 88 each side (176 total) 0011: 80 each side (160 total) 0100: 72 each side (144 total) 0101: 64 each side (128 total) 0110: 56 each side (112 tota)l 0111: 48 each side (96 total) 1000: 40 each side (80 total) 1001: 32 each side (64 total) 1010: 24 each side (48 total) 1011: 16 each side (32 total) 1100: 8 each side (16 total) Others Invalid Used to limit performance for tuning purposes. This defeature mode should not be used in conjuction with ctagentryavailmask in IRPMISCDFX2/IRPMISCDFX3. User must also insure that the Switch CSIPOOLDFX01 CSR maxcache fields is programmed accordingly to refect the acutal number of write cache entries used in IRP else unknown behavior may result. rrbsize3 is located at CIPCTRL16

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x140**				

Bit	Attr	Default	Description
8:6	RW	0x1	numrtid_vcp: 000: 0 001: 1 010: 2 011: 3 100: 4 Others: Reserved
5:3	RW	0x0	numrtids_vc1: 000: 0 001: 1 010: 2 011: 3 100: 4 Others: Reserved
2:2	RW	0x0	pcirdcurr_drduc_sel_vcp: 0: PCIRdCurrent 1: DRd.UC **Note:** This CSR should always be set to '0' due to Cbo issues in handling VCp requests as DRd.UC.
1:1	RW	0x0	diswrcomb: Causes all writes to send a WB request as soon as M-state is acquired. 0: Enable b2b Write Combining for writes from same port 1: Disable b2b Write Combining for writes from same port
0:0	RW	0x0	pcirdcurr_drduc_sel: On Inbound Coherent Reads selection of RdCur or DRd is done based on this configuration bit. 0: PCIRdCurrent 1: DRd.UC

7.4.37 cipsts

Coherent Interface Protocol Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x144**				

Bit	Attr	Default	Description
2:2	RO-V	0x1	rrb_non_phold_arb_empty: This indicates that there are no pending requests in the RRB with the exception of ProcLock/Unlock messages to the lock arbiter.0 - Pending RRB requests 1 - RRB Empty except for any pending Proclock/Unlock This is a live bit and hence can toggle clock by clock. This is provided mostly as a debug visibility feature.
1:1	RO-V	0x1	rrb_empty: This indicates that there are no pending requests in the RRB.0 - Pending RRB requests 1 - RRB Empty This is a live bit and hence can toggle clock by clock. This is provided mostly as a debug visibility feature.
0:0	RO-V	0x0	flush_pending_writes: This bit gets cleared whenever bit 31 in CPICTRL is written to 1 by software and gets set by hardware when the pending writes in the Write Cache (at the time bit 31 in CIPCTRL is written to 1 by software) complete that is, the Write Cache/RRB entry is deallocated for all those writes.

7.4.38 cipdcasad

Coherent Interface Protocol DCA Source Address Decode.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x148**				

Bit	Attr	Default	Description
31:29	RW	0x0	dcalt7: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 7
28:26	RW	0x0	dcalt6: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 6
25:23	RW	0x0	dcalt5: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 5
22:20	RW	0x0	dcalt4: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 4
19:17	RW	0x0	dcalt3: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 3
16:14	RW	0x0	dcalt2: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 2

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 0
Offset: 0x148

Bit	Attr	Default	Description
13:11	RW	0x0	dcalt1: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 1
10:8	RW	0x0	dcalt0: For a TPH/DCA request, specifies the target NodeID[2:0] when the inverted Tag[2:0] is 0
0:0	RW	0x0	dcaen: When disabled, PrefetchHint will not be sent on the coherent interface. 0: Disable TPH/DCA Prefetch Hints 1: Enable TPH/DCA Prefetch Hints **Notes:** • This register is locked based on DISDCA fuse • This table is programmed by the BIOS and this bit is set when the table is valid

7.4.39 cipintrc

Coherent Interface Protocol Interrupt Control.

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 0
Offset: 0x14c

Bit	Attr	Default	Description
25:25	RW	0x0	dis_intx_route2ich:
24:24	RW	0x0	route_nmi2mca:
18:18	RW	0x0	smi_msi_en:
17:17	RW	0x0	init_msi_en:
16:16	RW	0x0	nmi_msi_en:
13:13	RW-L	0x1	ferr_mask: **Note:** Locked by RSPLCK
12:12	RW	0x1	a20m_mask:
11:11	RW	0x1	intr_mask:
10:10	RW	0x1	smi_mask:
9:9	RW	0x1	init_mask:
8:8	RW	0x1	nmi_mask:
7:7	RW-L	0x0	ia32_or_ipf: **Note:** Locked by RSPLCK
1:1	RW	0x0	logical:
0:0	RW-L	0x0	cluster_check_sampling_mode: **Note:** Locked by RSPLCK

7.4.40 cipintrs

Coherent Interface Protocol Interrupt Status.

This register is to be polled by the BIOS to determine if internal pending system interrupts are drained out of IIO. General usage model is for software to quiesce the source for example, IOM global error logic of a system event like SMI, then poll this register till this register indicates that the event is not pending inside IIO. One additional read is required from software, after the register first reads 0 for the associated event.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x154**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	smi: This is set whenever IIO forwards a VLW from PCH that had the SMI bit assserted
30:30	RW1CS	0x0	nmi: This is set whenever IIO forwards a VLW from PCH that had the NMI bit assserted
7:7	RO-V	0x0	mca_ras_evt_pending:
6:6	RO-V	0x0	nmi_ras_evt_pending:
5:5	RO-V	0x0	smi_ras_evt_pending:
4:4	RO-V	0x0	intr_evt_pending:
3:3	RO-V	0x0	a20m_evt_pending:
2:2	RO-V	0x0	init_evt_pending:
1:1	RO-V	0x0	nmi_evt_pending:
0:0	RO-V	0x0	vlw_msgpend: either generated internally or externally

7.4.41 vtbar

Base Address Register for Intel VT-d.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x180**				

Bit	Attr	Default	Description
31:13	RW-LB	0x0	vtd_chipset_base_address: Provides an aligned 8K base address for IIO registers relating to Intel VT-d. All inbound accesses to this region are completer aborted by the IIO.
0:0	RW-LB	0x0	vtd_chipset_base_address_enable: **Note:** Accesses to registers pointed to by VTBAR are accessible by means of message channel or JTAG mini-port, irrespective of the setting of this enable bit that is, even if this bit is clear, read/write to Intel VT-d registers are completed normally (writes update registers and reads return the value of the register) for accesses from message channel or JTAG mini-port. This bit is RW-LB that is, lock is determined based on the 'trusted' bit in message channel when VTGENCTRL[15] is set, else it is RO.

7.4.42 vtgenctrl

Intel VT-d General Control.

Type: CFG **Bus:** 0 **Offset:** 0x184	**PortID:** N/A **Device:** 5	**Function:** 0	

Bit	Attr	Default	Description
15:15	RW-O	0x0	lockvtd: When this bit is 0, the VTBAR[0] is RW-LB, else it is RO.
7:4	RW-LB	0xa	hpa_limit: Represents the host processor addressing limit 0000: 2^36 (that is, bits 35:0) 0001: 2^37 (that is, bits 36:0) ... 1010: 2^46 (that is, bits 45:0) When Intel VT-d translation is enabled on an Intel VT-d engine, all host addresses (during page walks) that go beyond the limit specified in this register will be aborted by IIO. **Note:** Pass-through and 'translated' ATS accesses carry the host-address directly in the access and are subject to this check as well.
3:0	RW-LB	0x8	gpa_limit: Represents the guest virtual addressing limit for the non-Isoch Intel VT-d engine. 0000: 2^40 (that is, bits 39:0) 0001: 2^41 (that is, bits 40:0) .. 0111: 2^47 1000: 2^48 Others: Reserved When Intel VT-d translation is enabled, all incoming guest addresses from PCI Express*, associated with the non-isoch Intel VT-d engine, that go beyond the limit specified in this register will be aborted by IIO and a UR response returned. This register is not used when translation is not enabled. **Note**: 'Translated' and 'pass-through' addresses are in the 'host-addressing' domain and NOT 'guest-addressing' domain and hence GPA_LIMIT checking on those accesses are bypassed and instead HPA_LIMIT checking applies.

7.4.43 vtgenctrl2

Intel VT-d General Control 2.

Type: CFG **Bus:** 0 **Offset:** 0x18c		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
18:12	RW-LB	0x4	tlb_free_entry_limit:
11:11	RW-LB	0x0	lructrl: Controls what increments the LRU counter that is used to degrade the LRU bits in the IOTLB, L1/L2, and L3 caches. 1: Count Cycles same as TB 0: Count Requests
10:7	RW-LB	0x7	lt: Controls the rate at which the LRU buckets should degrade. If we are in "Request" mode (LRUCTRL = 0), then we will degrade LRU after 16 * N requests where N is the value of this field. If we are in "Cycles" mode (CRUCTRL = 1), then we will degrade LRU after 256 * N cycles where N is the value of this field. The default value of 0x7 along with LRUCTRL0 will give us a default behavior of decreasing the LRU buckets every 112 requests.
6:5	RW-LB	0x1	prefetch_control: Queued invalidation, interrupt table read, context table reads and root table reads NEVER have prefetch/snarf/reuse capability. This is a general rule. Beyond that the Prefetch Control bits control additional behavior as shown below. This field controls which Intel VT-d reads are to be considered for prefetchsnarfreuse. 00: Prefetch/snarf/reuse is turned off that is, IRP cluster never reuses the Intel VT-d read data 01: Prefetch/snarf/reuse is enabled for all leafnon-leaf Intel VT-d page walk reads. 10: Prefetch/snarf/reuse is enabled only on leaf not non-leaf Intel VT-d page walks reads with CC.ALH bit set 11: Prefetch/snarf/reuse is enabled on ALL leaf not non-leaf Intel VT-d page walks reads regardless of the setting of the CC.ALH bit
3:3	RW-LB	0x0	ignoreubitleafeviction:
2:2	RW-LB	0x0	evictnonleafat01:
1:1	RW-LB	0x0	dontevictleafat01:

7.4.44 iotlbpartition

IOTLB Partitioning Control.

Type: CFG **Bus:** 0 **Offset:** 0x194		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
28:27	RW	0x0	rangesel_dmi_20_22:
26:25	RW	0x0	rangesel_iou24_upper_x2:
24:23	RW	0x0	rangesel_iou23_upper_x2:
14:13	RW	0x0	rangesel_me:

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x194**				

Bit	Attr	Default	Description
12:11	RW	0x0	rangesel_cb:
10:9	RW	0x0	rangesel_intr:
0:0	RW-LB	0x0	iotlb_parten: 0: Disabled 1: Enabled

7.4.45 vtuncerrsts

Intel VT-d Uncorrectable Error Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x1a8**				

Bit	Attr	Default	Description
31:31	RW1CS	0x0	vtderr: When set, this bit is set when an Intel VT-d specification defined error has been detected (and logged in the Intel VT-d fault registers)
8:8	RW1CS	0x0	protmemviol:
7:7	RW1CS	0x0	miscerrs: Illegal request to 0xFEE, GPAHPA limit error status
6:6	RW1CS	0x0	unsucc_ci_rdcp:
5:5	RW1CS	0x0	perr_tlb1:
4:4	RW1CS	0x0	perr_tlb0:
3:3	RW1CS	0x0	perr_l3_lookup:
2:2	RW1CS	0x0	perr_l2_lookup:
1:1	RW1CS	0x0	perr_l1_lookup:
0:0	RW1CS	0x0	perr_context_cache:

7.4.46 vtuncerrmsk

Intel VT-d Uncorrectable Error Mask.

Mask out error reporting to IIO. Bit 31 should always be set to 1. We recommend that the other bits be left as zero so these internal errors are reported out.

Setting bits will not prevent any error collecting INSIDE of Intel VT-d in the Intel VT-d Fault Recording Registers.

Type: CFG **Bus:** 0 **Offset:** 0x1ac		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:31	RWS	0x1	vtderr_msk: This bit should be set to 1 by the BIOS. It is highly recommended that this bit is never set to 0. If Intel VT-d errors are configured to be fatal, leaving this bit set to 0 will cause Fatal errors to be reported when devices send illegal requests. This is generally undesireable.
8:8	RWS	0x0	protmemviol_msk:
7:7	RWS	0x0	miscerrm: Illegal request to 0xFEE, GPAHPA limit error mask
6:6	RWS	0x0	unsucc_ci_rdcp_msk:
5:5	RWS	0x0	perr_tlb1_msk:
4:4	RWS	0x0	perr_tlb0_msk:
3:3	RWS	0x0	perr_l3_lookup_msk:
2:2	RWS	0x0	perr_l2_lookup_msk:
1:1	RWS	0x0	perr_l1_lookup_msk:
0:0	RWS	0x0	perr_context_cache_msk:

7.4.47 vtuncerrsev

Intel VT-d Uncorrectable Error Severity.

Type: CFG **Bus:** 0 **Offset:** 0x1b0		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:31	RWS	0x0	vtderr_sev: When set, this bit escalates reporting of Intel VT-d specification defined errors, as FATAL errors. When clear, those errors are escalated as Nonfatal errors. Setting this bit to a 1 can allow a guest VM to trigger an unrecoverable FATAL error at the platform. It is HIGHLY recommended that the BIOS keep this bit set to 0, as such behavior is generally undesirable.
8:8	RWS	0x1	protmemviol_sev:
7:7	RWS	0x1	miscerrsev: Illegal request to 0xFEE, GPAHPA limit error severity
6:6	RWS	0x0	unsucc_ci_rdcp_sev:
5:5	RWS	0x1	perr_tlb1_sev:

Type: **CFG** **Bus:** **0** **Offset:** **0x1b0**		**PortID:** **N/A** **Device:** **5**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
4:4	RWS	0x1	perr_tlb0_sev:
3:3	RWS	0x1	perr_l3_lookup_sev:
2:2	RWS	0x1	perr_l2_lookup_sev:
1:1	RWS	0x1	perr_l1_lookup_sev:
0:0	RWS	0x1	perr_context_cache_sev:

7.4.48 vtuncerrptr

Intel VT-d Uncorrectable Error Pointer.

Type: **CFG** **Bus:** **0** **Offset:** **0x1b4**		**PortID:** **N/A** **Device:** **5**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
4:0	ROS_V	0x0	vt_uncferr_ptr: This field points to which of the unmasked uncorrectable errors happened first. This field is only valid when the corresponding error is unmasked and the status bit is set and this field is rearmed to load again when the status bit indicated to by this pointer is cleared by software from 1 to 0. Value of 0x0 corresponds to bit 0 in VTUNCERRSTS register, value of 0x1 corresponds to bit 1 and so forth.

7.4.49 iiomiscctrl

IIO MISC Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x1c0**				

Bit	Attr	Default	Description
42:42	RW-LB	0x0	enable_pcc_eq0_sev1:
41:41	RW	0x0	en_poismsg_spec_behavior: A received poison packet is treated as a Fatal error if it's severity bit is set, but treated as a correctable if the severity bit is cleared and logged in both the UNCERRSTS register and the Advisory Non-Fatal Error bit in the CORERRSTS register. In the processor B0, a POISFEN bit forces the poison error to be logged as an Advisory Non-Fatal error. When this bit is set, the poison severity bit can force Fatal behavior regardless of POISFEN. Generally, however, MCA needs to have priority over AER drivers, so this bit default is 0. **Note:** The PCIe specification requires this bit to be 0. When this bit is clear: sev pfen error 0 0 non-fatal 0 1 correctable 1 0 fatal 1 1 correctable When this bit is set: sev pfen error 0 0 non-fatal 0 1 correctable 1 0 fatal 1 1 fatal
40:40	RW	0x0	enable_io_mca:
39:39	RW	0x0	disable_new_apic_ordering: When this bit is set, behavior returns to the original behavior.
38:38	RWS_O	0x1	uniphy_en_fuse4_pwrdn:
37:37	RW	0x0	poisfen: Enables poisoned data received inbound (either inbound posted data or completions for outbound reads that have poisoned data) to be forwarded to the destination (DRAM or Cache or PCIe Peer). 0: Poison indication is not forwarded with the data (this may result in silent corruption if AER poison reporting is disabled.) 1: Poison indication is forwarded with the data (this may result in a conflict with MCA poison reporting if AER poison reporting is enabled)

Type: **CFG**	**PortID:** **N/A**		
Bus: **0**	**Device:** **5**	**Function:** **0**	
Offset: **0x1c0**			

Bit	**Attr**	**Default**	**Description**
34:32	RWS	0x0	showportid: A Port Identifier that identifies which PCI Express* port a transaction comes from will be placed in the AD Ring TNID[2:0] field of the request packet, when enabled. This field is normally used for DCAHint and is not used for normal demand read. Since there are up to 11 specific ports, then Port ID is encoded in 4 bits. Only three bits can be selected to be sent in TNID as follows: 100: TNID[2:0] = PortID[3:1] 011: TNID[2:0] = PortID[3:2, 0] 010: TNID[2:0] = PortID[3, 1:0] 001: TNID[2:0] = PortID[2:0] 000: IIO will not send Port ID information in the TNID[2:0] field The PortIDs are mapped as follows: 0: Device 0 Function 0 DMIPCIe port 0 (IOU2) 1: Device 1 Function 0 Port 1a x4 or x8 (IOU2) 2: Device 1 Function 1 Port 1b x4 (IOU2) 3: Device 2 Function 0 Port 2a x4, x8, or x16 (IOU0) 4: Device 2 Function 1 Port 2b x4 (IOU0) 5: Device 2 Function 2 Port 2c x4 or x8 (IOU0) 6: Device 2 Function 3 Port 2d x4 (IOU0) 7: Device 3 Function 0 Port 3a x4, x8, or x16 8: Device 3 Function 1 Port 3b x4 (IOU1) 9: Device 3 Function 2 Port 3c x4 or x8 (IOU1) 10: Device 3 Function 3 Port 3d x4 (IOU1) 11: CB 12: Intel VT
30:30	RW	0x1	treat_last_write_in_descriptor_specially: Treat CB DMA writes with NS = RO = 1 NS is enabled in CB DMA 'last write in descriptor', as-if NS = 1 and RO = 0 write
25:25	RWS	0x1	cballocen: When set, use Allocating Flows for non-DCA writes from CB DMA. This bit does not affect DCA requests when DCA requests are enabled (bit 21 of this register). A DCA request is identified as matching the DCA requestor ID and having a Tag of non-zero. All DCA requests are always allocating, unless they are disabled, or unless all allocating flows are disabled (bit 24). If all allocating flows are disabled, then DCA requests are also disabled. The BIOS is to leave this bit at default of 1b for all but DMI port.
24:24	RW	0x0	disable_all_allocating_flows: When this bit is set, IIO will no more issue any new inbound IDI command that can allocate into LLC. Instead, all the writes will use one of the non-allocating commands - PCIWiL/PCIWiLF/PCINSWr/PCINSWrF.This is provided primarily for PSMI where we need a mode to not allocate into the LLC. Software should set this bit only when no requests are being actively issued on IDI. So either a lock/quiesce flow should be employed before this bit is set/cleared or it should be set up before DMA is enabled in system.
22:22	RW	0x0	disable_ro_on_writes_from_cb_dma:
20:20	RW	0x0	switch_arbitration_weight_for_CB_DMA: 1

Type: **CFG** **Bus:** **0** **Offset:** **0x1c0**		**PortID:** **N/A** **Device:** **5**	**Function:** **0**
Bit	**Attr**	**Default**	**Description**
19:19	RW	0x0	rvgaen: Remote VGA EnableEnables VGA accesses to be sent to remote node. If set, accesses to the VGA region (A_0000 to B_FFFF) will be forwarded to the CBo where it will determine the node ID where the VGA region resides. It will then be forwarded to the given remote node. If clear, then VGA accesses will be forwarded to the local PCIe port that has it's VGAEN set. If none have their VGAEN set, then the request will be forwarded to the local DMI port, if operating in DMI mode. If it is not operating in DMI mode, then the request will be aborted.
18:18	RW	0x1	disable_inbound_ro_for_vc0: When enabled this mode will treat all inbound write traffic as RO = 0 for VC0. This affects all PCI Express* ports and the DMI port.0 - Ordering of inbound transactions is based on RO bit for VC0 1 - RO bit is treated as '0' for all inbound VC0 traffic **Note:** This impacts only the NS write traffic because for snooped traffic RO bit is ignored by h/w. When this bit is set, the NS write if enabled BW is going to be generally bad. **Note:** This bit does not impact VC1 and VCm writes
17:16	RW	0x1	dmi_vc1_write_ordering: Mode is used to control VC1 write traffic from DMI (Intel VT). 00: Reserved 01: Serialize writes on CSI issuing one at a time 10: Pipeline writes on CSI except for writes with Tag value of 0x21 which are issued only after prior writes have all completed and reached global observability 11: Pipeline writes on CSI based on RO bit that is, if RO = 1, pipeline without waiting for prior write to have reached global observability. If RO0, then it needs to wait till prior writes have all reached global observability.
15:15	RW	0x0	dmi_vc1_vt_d_fetch_ordering: This mode is to allow VC1 Intel VT-d conflicts with outstanding VC0 Intel VT-d reads on IDI to be pipelined. This can occur when the Intel VT-d tables are shared between Intel VT (VC1) and other devices. To ensure QoS the Intel VT-d reads from VC1 need to be issued in parallel with non-Isoc accesses to the same cacheline. 0: Serialize all IDI address conflicts to DRAM 1: Pipeline Intel VT-d reads from VC1 with address conflict on IDI **Note:** A maximum of 1 VC1 Intel VT-d read and 1 non-VC1 Intel VT-d read to the same address can be outstanding on IDI.
14:14	RW	0x0	pipeline_ns_writes_on_csi: When set, allows inbound non-snooped writes to pipeline at the coherent interface - issuing the writes before previous writes are completed in the coherent domain.
13:13	RW	0x0	vc1_reads_bypass_writes: 0: VC1 Reads push VC1 writes 1: VC1 Reads are allowed to bypass VC1 writes
12:12	RW	0x0	lock_thaw_mode: Mode controls how inbound queues in the south agents (PCIE, DMI) thaw when they are target of a locked read. See xref for details on when this should be used and on the restrictions in its use. 0: Thaw only posted requests 1: Thaw posted and non-posted requests. **Note:** The lock target is also a 'problematic' port (as indicated by bit 38 in MISCCTRLSTS register), then this becomes meaningless because both posted and non-posted requests are thawed.

Type: CFG Bus: 0 Offset: 0x1c0		PortID: N/A Device: 5	Function: 0
Bit	**Attr**	**Default**	**Description**
10:10	RW	0x0	legacy_port: Sockets where the NodeID = 0 are generally identified as having the legacy DMI port. But there is still a possibility that another socket also has a NodeID = 0. The system is configured by software to route legacy transactions to the correct socket. However, inbound legacy messages received on a PCIe port of a socket with NodeID = 0 that is not the true legacy port need to be routed to a remote socket that is the true legacy port. For a local NodeID is zero, this bit is used to determine if inbound messages should be routed to a DMI port on a remote socket with NodeID = 0, or if the messages should be sent to the local DMI port, since the local NodeID is also 0. If the local NodeID is not zero, then this bit is ignored. 0: indicates this socket has the true DMI legacy port, send legacy transactions to local DMI port 1: indicates this is a non-legacy socket, send legacy transactions to the Coherent Interface **Notes:** • This bit does not affect routing for non-message transactions. It only affects inbound messages that need to be routed to the true legacy port. • This bit is NOT used for any outbound address decode routing purposes. Outbound traffic that is subtractively decoded will always be forwarded to local DMI port, if one exists, or it will be aborted. • The default value of this field is based on the NodeID and FWAGENT_DMIMODE straps. • Software can only change this bit after reset during early boot phase, but must guarantee there is no traffic flowing through the system, except for the write that changes this bit.
9:9	RW	0x1	Reserved.
8:8	RW	0x0	tocmvalid: Enables the TOCM field.
7:3	RW	0xe	tocm: Indicates the top of Core physical addressability limit. 00000-00100: Reserved 00101: 2^37 00110: 2^38 ... 1110: 2^46 01111 -11111: Reserved iio uses this to abort all inbound transactions that cross this limit.
2:2	RW	0x0	en1k: This bit when set, enables 1K granularity for I/O space decode in each of the virtual P2P bridges corresponding to root ports, and DMI ports.
1:1	RWS_O	0x0	uniphy_disable: Place entire UNIPHY in L2 for when no ports are used, as in some multi-socket configurations
0:0	RW-LB	0x0	enable_isa_hole: When this bit is set, inbound DMA accesses to the ISA Hole region are aborted by IIO. If clear, inbound DMA accesses to the ISA hole region are forwarded to dram. The ISA Hole is no longer supported by the processor. This bit must never be set.

7.4.50 ltdpr

LT DMA Protected Range.

General Description: This register holds the address and size of the DMA protected memory region for LT-SX MP usage.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x290**				

Bit	Attr	Default	Description
31:20	RO-V	0x0	topofdpr: Top address + 1 of DPR. This is RO, and it is copied by hardware from TSEGBASE[31:20].
11:4	RW-L	0x0	size: This is the size of memory, in MB, that will be protected from DMA accesses. A value of 0x00 in this field means no additional memory is protected. The maximum amount of memory that will be protected is 255MB. The amount of memory reported in this field will be protected from all DMA accesses. The top of the protected range is typically the BASE of TSEG -1. The BIOS is expected to program that in to bits 31:20 of this register. **Notes:** • If TSEG is not enabled, then the top of this range becomes the base ME stolen space, whichever would have been the location of TSEG, assuming it had been enabled. • The DPR range works independently of any other range - Generic Protected ranges, TSEG range, Intel VT-d tables, Intel VT-d protection ranges, MMCFG protection range and is done post any VTd translation or Intel TXT checks. Therefore incoming cycles are checked against this range after the VTd translation and faulted if they hit this protected range, even if they passed the VTd translation. • All the memory checks are OR'ed with respect to NOT being allowed to go to memory. So if either Generic protection range, DPR, Intel VT-d, TSEG range disallows the cycle, then the cycle is not allowed to go to memory. Or in other words, all the above checks must pass before a cycle is allowed to DRAM. • DMA remap engines are allowed to access the DPR region without any faulting. It is always legal for any DMA remap engine to read or write into the DPR region, thus DMA remap accesses must not be checked against the DPR range.
2:2	RW-L	0x0	commandbit: Writing a '1' to this bit will enable protection. Writing a '0' to this bit will disable protection.
1:1	RO	0x0	protregsts: IIO sets this bit when the protection has been enabled in hardware and for all practical purposes this should be immediate. When protection is disabled, then this bit is clear
0:0	RW-O	0x0	lock: Bits 19:0 are locked down in this register when this bit is set. Can this be set while other bits are being written to in the same write transaction

7.4.51 lcfgbus_base

Type: CFG **Bus:** 0 **Offset:** 0x41c		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
7:0	RW	0x0	lcfgbus_base:

7.4.52 lcfgbus_limit

Type: CFG **Bus:** 0 **Offset:** 0x41d		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
7:0	RW	0x0	lcfgbus_limit:

7.4.53 csipintrs

Type: CFG **Bus:** 0 **Offset:** 0x450		**PortID:** N/A **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
7:7	RO-V	0x0	mca_ras_evt_pend:
6:6	RO-V	0x0	nmi_ras_evt_pend:
5:5	RO-V	0x0	smi_ras_evt_pend:
4:4	RO-V	0x0	intr_evt_pend:
3:3	RO-V	0x0	a20m_evt_pend:
2:2	RO-V	0x0	init_evt_pend:
1:1	RO-V	0x0	nmi_evt_pend:
0:0	RO-V	0x0	smi_evt_pend:

7.5 Device 5 Function 0 MMIO Region VTBAR

Intel VT-d registers are all addressed using aligned Dword or aligned Qword accesses. Any combination of bits is allowed within a Dword or Qword access. The Intel VT-d remap engine registers corresponding to the port represented by Device 0, occupy the first 4 K of offset starting from the base address defined by VTBAR register.

Table 7-8. Integrated I/O Device 5 Function 0 MMIO Region VTBAR Register Address Map (Sheet 1 of 2)

Register Name	Offset	Size
vtd0_version	0x0	32
vtd0_cap	0x8	64
vtd0_ext_cap	0x10	64
vtd0_glbcmd	0x18	32
vtd0_glbsts	0x1c	32
vtd0_rootentryadd	0x20	64
vtd0_ctxcmd	0x28	64
vtd0_fltsts	0x34	32
nonisoch_fltevtctrl	0x38	32
nonisoch_fltevtdata	0x3c	32
vtd0_fltevtaddr	0x40	32
vtd0_fltevtupraddr	0x44	32
vtd0_pmen	0x64	32
vtd0_prot_low_mem_base	0x68	32
vtd0_prot_low_mem_limit	0x6c	32
vtd0_prot_high_mem_base	0x70	64
vtd0_prot_high_mem_limit	0x78	64
vtd0_inv_queue_head	0x80	64
vtd0_inv_queue_tail	0x88	64
vtd0_inv_queue_add	0x90	64
vtd0_inv_comp_status	0x9c	32
nonisoch_inv_cmp_evtctrl	0xa0	32
nonisoch_invevtdata	0xa4	32
vtd0_inv_comp_evt_addr	0xa8	32
vtd0_inv_comp_evt_upraddr	0xac	32
vtd0_intr_remap_table_base	0xb8	64
vtd0_fltrec0_gpa	0x100	64
vtd0_fltrec0_src	0x108	64
vtd0_fltrec1_gpa	0x110	64
vtd0_fltrec1_src	0x118	64
vtd0_fltrec2_gpa	0x120	64
vtd0_fltrec2_src	0x128	64
vtd0_fltrec3_gpa	0x130	64
vtd0_fltrec3_src	0x138	64

Table 7-8. Integrated I/O Device 5 Function 0 MMIO Region VTBAR Register Address Map (Sheet 2 of 2)

Register Name	Offset	Size
vtd0_fltrec4_gpa	0x140	64
vtd0_fltrec4_src	0x148	64
vtd0_fltrec5_gpa	0x150	64
vtd0_fltrec5_src	0x158	64
vtd0_fltrec6_gpa	0x160	64
vtd0_fltrec6_src	0x168	64
vtd0_fltrec7_gpa	0x170	64
vtd0_fltrec7_src	0x178	64
vtd0_invaddrreg	0x200	64
vtd0_iotlbinv	0x208	64
vtd1_version	0x1000	32
vtd1_cap	0x1008	64
vtd1_ext_cap	0x1010	64
vtd1_glbcmd	0x1018	32
vtd1_glbsts	0x101c	32
vtd1_rootentryadd	0x1020	64
vtd1_ctxcmd	0x1028	64
vtd1_fltsts	0x1034	32
vtd1_fltevtaddr	0x1040	32
vtd1_fltevtupraddr	0x1044	32
vtd1_pmen	0x1064	32
vtd1_prot_low_mem_base	0x1068	32
vtd1_prot_low_mem_limit	0x106c	32
vtd1_prot_high_mem_base	0x1070	64
vtd1_prot_high_mem_limit	0x1078	64
vtd1_inv_queue_head	0x1080	64
vtd1_inv_queue_tail	0x1088	64
vtd1_inv_queue_add	0x1090	64
vtd1_inv_comp_status	0x109c	32
vtd1_inv_comp_evt_addr	0x10a8	32
vtd1_inv_comp_evt_upraddr	0x10ac	32
vtd1_intr_remap_table_base	0x10b8	64
vtd1_fltrec0_gpa	0x1100	64
vtd1_fltrec0_src	0x1108	64
vtd1_invaddrreg	0x1200	64
vtd1_iotlbinv	0x1208	64

7.5.1 vtd[0:1]_version

Intel VT-d Version Number.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x0, 0x1000**				

Bit	Attr	Default	Description
7:4	RO	0x1	major_revision:
3:0	RO	0x0	minor_revision:

7.5.2 vtd[0:1]_cap

Intel VT-d Capabilities.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x8, 0x1008**				

Bit	Attr	Default	Description
55:55	RO	0x1	dma_read_draining: The processor supports hardware based draining
54:54	RO	0x1	dma_write_draining: The processor supports hardware based write draining
53:48	RO	0x12	mamv: The processor support MAMV value of 12h (up to 1G super pages).
47:40	RO	0x7	number_of_fault_recording_registers: The processor supports 8 fault recording registers
39:39	RO	0x1	page_selective_invalidation: Supported in IIO
37:34	RW-O	0x3	super_page_support: 2MB, 1GB supported.
33:24	RO	0x10	fault_recording_register_offset: Fault registers are at offset 100h
23:23	RO	0x0	spatial_separation:
22:22	RO	0x1	zlr: Zero-length DMA requests to write-only pages supported.
21:16	RO-V	0x2f	mgaw: This register is set by the processor-based on the setting of the GPA_LIMIT register. The value is the same for both the VT and non-VT engines. This is because the translation for VT has been extended to be 4-level (instead of 3).
12:8	RO	0x4	sagaw: Supports 4-level walk on both VT and non-VT engines
7:7	RO	0x0	tcm: The processor does not cache invalid pages. This bit should always be set to 0 on hardware. It can be set to one when we are doing software virtualization of Intel VT-d.
6:6	RO	0x1	phmr_support: The processor supports protected high memory range
5:5	RO	0x1	plmr_support: The processor supports protected low memory range \

Type: MEM **PortID:** 8’h7e
Bus: 0 **Device:** 5 **Function:** 0
Offset: 0x8, 0x1008

Bit	Attr	Default	Description
4:4	RO	0x0	rwbf:
3:3	RO	0x0	advanced_fault_logging: The processor does not support advanced fault logging
2:0	RO	0x6	number_of_domains_supported: The processor supports 256 domains with 8-bit domain ID

7.5.3 vtd[0:1]_ext_cap

Extended Intel VT-d Capability.

Type: MEM **PortID:** 8’h7e
Bus: 0 **Device:** 5 **Function:** 0
Offset: 0x10, 0x1010

Bit	Attr	Default	Description
23:20	RO	0xf	maximum_handle_mask_value: IIO supports all 16 bits of handle being masked. **Note:** IIO always performs global interrupt entry invalidation on any interrupt cache invalidation command and h/w never really looks at the mask value.
17:8	RO	0x20	invalidation_unit_offset: IIO has the invalidation registers at offset 200h
7:7	RO	0x1	snoop_control: 0: Hardware does not support 1-setting of the SNP field in the page-table entries.1: Hardware supports the 1-setting of the SNP field in the page-table entries. IIO supports snoop override only for the non-isoch Intel VT-d engine
6:6	RO	0x1	pass_through: IIO supports pass through. This bit is RW-O for defeaturing in case of post-si bugs.
4:4	RW-O	0x1	ia32_extended_interrupt_mode: IIO supports the extended interrupt mode
3:3	RO	0x1	interrupt_remapping_support: IIO supports this
2:2	RW-O	0x1	device_tlb_support: IIO supports ATS for the non-isoch Intel VT-d engine. This bit is RW-O for non-isoch engine in case we might have to defeature ATS post-si. For VTD[0]_EXT_CAP.Bit[2] the default is 1, but can be programmed to 0. Clarification: For VTD[1]_EXT_CAP.Bit[2] the default is 0
1:1	RO	0x1	queued_invalidation_support: IIO supports this For VTD[1]_EXT_CAP.Bit[1] the default is 0.
0:0	RW-O	0x0	coherency_support: The BIOS can write to this bit to indicate to hardware to either snoop or not-snoop the DMA/Interrupt table structures in memory (root/context/pd/pt/irt). **Note:** This bit is expected to be always set to 0 for the Intel VT-d engine and programmability is only provided for that engine for debug reasons.

7.5.4 vtd[0:1]_glbcmd

Intel VT-d Global Command.

Type: MEM **Bus:** 0 **Offset:** 0x18, 0x1018		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:31	RW	0x0	translation_enable: Software writes to this field to request hardware to enable/disable DMA-remapping hardware.0: Disable DMA-remapping hardware 1: Enable DMA-remapping hardware Hardware reports the status of the translation enable operation through the TES field in the Global Status register. Before enabling (or re-enabling) DMA-remapping hardware through this field, software must: - Setup the DMA-remapping structures in memory - Flush the write buffers (through WBF field), if write buffer flushing is reported as required. - Set the root-entry table pointer in hardware (through SRTP field). - Perform global invalidation of the context-cache and global invalidation of IOTLB - If advanced fault logging supported, setup fault log pointer (through SFL field) and enable advanced fault logging (through EAFL field). There may be active DMA requests in the platform when software updates this field. Hardware must enable or disable remapping logic only at deterministic transaction boundaries, so that any in-flight transaction is either subject to remapping or not at all.
30:30	RW-V	0x0	set_root_table_pointer: Software sets this field to set/update the root-entry table pointer used by hardware. The root-entry table pointer is specified through the Root-entry Table Address register.Hardware reports the status of the root table pointer set operation through the RTPS field in the Global Status register. The root table pointer set operation must be performed before enabling or re-enabling (after disabling) DMA remapping hardware. After a root table pointer set operation, software must globally invalidate the context cache followed by global invalidate of IOTLB. This is required to ensure hardware uses only the remapping structures referenced by the new root table pointer, and not any stale cached entries. While DMA-remapping hardware is active, software may update the root table pointer through this field. However, to ensure valid in-flight DMA requests are deterministically remapped, software must ensure that the structures referenced by the new root table pointer are programmed to provide the same remapping results as the structures referenced by the previous root table pointer. Clearing this bit has no effect.
29:29	RO	0x0	set_fault_log_pointer:
28:28	RO	0x0	enable_advanced_fault_logging:
27:27	RO	0x0	write_buffer_flush:
26:26	RW	0x0	queued_invalidation_enable: Software writes to this field to enable queued invalidations.0: Disable queued invalidations. In this case, invalidations must be performed through the Context Command and IOTLB Invalidation Unit registers. 1: Enable use of queued invalidations. Once enabled, all invalidations must be submitted through the invalidation queue and the invalidation registers cannot be used till the translation has been disabled. The invalidation queue address register must be initialized before enabling queued invalidations. Also software must make sure that all invalidations submitted prior by means of the register interface are all completed before enabling the queued invalidation interface. Hardware reports the status of queued invalidation enable operation through QIES field in the Global Status register. Value returned on read of this field is undefined.

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x18, 0x1018**				

Bit	Attr	Default	Description
25:25	RW	0x0	interrupt_remapping_enable: 0: Disable Interrupt Remapping Hardware1: Enable Interrupt Remapping Hardware Hardware reports the status of the interrupt-remap enable operation through the IRES field in the Global Status register. Before enabling (or re-enabling) Interrupt-remapping hardware through this field, software must: - Setup the interrupt-remapping structures in memory - Set the Interrupt Remap table pointer in hardware (through IRTP field). - Perform global invalidation of IOTLB There may be active interrupt requests in the platform when software updates this field. Hardware must enable or disable remapping logic only at deterministic transaction boundaries, so that any in-flight interrupts are either subject to remapping or not at all. IIO must drain any in-flight translated DMA read/write, MSI interrupt requests queued within the root complex before completing the translation enable command and reflecting the status of the command through the IRES field in the GSTS_REG. Value returned on read of this field is undefined.
24:24	RW-V	0x0	set_interrupt_remap_table_pointer: Software sets this field to set/update the interrupt remapping table pointer used by hardware. The interrupt remapping table pointer is specified through the Interrupt Remapping Table Address register.Hardware reports the status of the interrupt remapping table pointer set operation through the IRTPS field in the Global Status register. The interrupt remap table pointer set operation must be performed before enabling or re-enabling (after disabling) interrupt remapping hardware through the IRE field. After an interrupt remap table pointer set operation, software must globally invalidate the interrupt entry cache. This is required to ensure hardware uses only the interrupt remapping entries referenced by the new interrupt remap table pointer, and not any stale cached entries. While interrupt remapping is active, software may update the interrupt remapping table pointer through this field. However, to ensure valid in-flight interrupt requests are deterministically remapped, software must ensure that the structures referenced by the new interrupt remap table pointer are programmed to provide the same remapping results as the structures referenced by the previous interrupt remap table pointer. Clearing this bit has no effect. IIO hardware internally clears this field before the 'set' operation requested by software has take effect.
23:23	RW	0x0	cfi: Compatibility Format Interrupt Software writes to this field to enable or disable Compatibility Format interrupts on Intel® 64 platforms. The value in this field is effective only when interrupt-remapping is enabled and Legacy Interrupt Mode is active. 0: Block Compatibility format interrupts. 1: Process Compatibility format interrupts as pass-through (bypass interrupt remapping). Hardware reports the status of updating this field through the CFIS field in the Global Status register. This field is not implemented on Itanium® platforms.

7.5.5 vtd[0:1]_glbsts

Intel VT-d Global Status.

Type: MEM **Bus:** 0 **Offset:** 0x1c, 0x101c			**PortID:** 8'h7e **Device:** 5 **Function:** 0
Bit	**Attr**	**Default**	**Description**
31:31	RO-V	0x0	translation_enable_status: When set, indicates that translation hardware is enabled and when clear indicates the translation hardware is not enabled.
30:30	RO-V	0x0	set_root_table_pointer_status: This field indicates the status of the root- table pointer in hardware.This field is cleared by hardware when software sets the SRTP field in the Global Command register. This field is set by hardware when hardware finishes the set root-table pointer operation (by performing an implicit global invalidation of the context-cache and IOTLB, and setting/updating the root-table pointer in hardware with the value provided in the Root-Entry Table Address register).
29:29	RO	0x0	set_fault_log_pointer_status:
28:28	RO	0x0	advanced_fault_logging_status:
27:27	RO	0x0	write_buffer_flush_status:
26:26	RO-V	0x0	queued_invalidation_interface_status: IIO sets this bit once it has completed the software command to enable the queued invalidation interface. Till then this bit is 0.
25:25	RO-V	0x0	interrupt_remapping_enable_status: OH sets this bit once it has completed the software command to enable the interrupt remapping interface. Till then this bit is 0.
24:24	RO-V	0x0	interrupt_remapping_table_pointer_status: This field indicates the status of the interrupt remapping table pointer in hardware. This field is cleared by hardware when software sets the SIRTP field in the Global Command register. This field is set by hardware when hardware completes the set interrupt remap table pointer operation using the value provided in the Interrupt Remapping Table Address register.
23:23	RO-V	0x0	cfis: Compatibility Format Interrupt Status The value reported in this field is applicable only when interrupt-remapping is enabled and Legacy interrupt mode is active. 0: Compatibility format interrupts are blocked. 1: Compatibility format interrupts are processed as pass-through (bypassing interrupt remapping).

7.5.6 vtd[0:1]_rootentryadd

Intel VT-d Root Entry Table Address.

Type: MEM **Bus:** 0 **Offset:** 0x20, 0x1020			**PortID:** 8'h7e **Device:** 5 **Function:** 0
Bit	**Attr**	**Default**	**Description**
63:12	RW	0x0	root_entry_table_base_address: 4K aligned base address for the root entry table. Software specifies the base address of the root-entry table through this register, and enables it in hardware through the SRTP field in the Global Command register. Reads of this register returns value that was last programmed to it.

7.5.7 vtd[0:1]_ctxcmd

Intel VT-d Context Command.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x28, 0x1028**				

Bit	Attr	Default	Description
63:63	RW-V	0x0	icc: Invalidate Context Entry Cache Software requests invalidation of context-cache by setting this field. Software must also set the requested invalidation granularity by programming the CIRG field. Software must read back and check the ICC field to be clear to confirm the invalidation is complete. Software must not update this register when this field is set. Hardware clears the ICC field to indicate the invalidation request is complete. Hardware also indicates the granularity at which the invalidation operation was performed through the CAIG field. Software must not submit another invalidation request through this register while the ICC field is set.Software must submit a context cache invalidation request through this field only when there are no invalidation requests pending at this DMA-remapping hardware unit. Since information from the context-cache may be used by hardware to tag IOTLB entries, software must perform domain-selective (or global) invalidation of IOTLB after the context cache invalidation has completed.
62:61	RW	0x0	cirg: Context Invalidation Request Granularity When requesting hardware to invalidate the context-entry cache (by setting the ICC field), software writes the requested invalidation granularity through this field.Following are the encoding for the 2-bit IRG field. 00: Reserved. Hardware ignores the invalidation request and reports invalidation complete by clearing the ICC field and reporting 00 in the CAIG field. 01: Global Invalidation request. 10: Domain-selective invalidation request. The target domain-id must be specified in the DID field. 11: Device-selective invalidation request. The target SID must be specified in the SID field, and the domain-id (programmed in the context-entry for this device) must be provided in the DID field. The processor aliases the h/w behavior for this command to the ‘Domain-selective invalidation request’. Hardware indicates completion of the invalidation request by clearing the ICC field. At this time, hardware also indicates the granularity at which the actual invalidation was performed through the CAIG field.
60:59	RO-V	0x0	caig: Context Actual Invalidation Granularity Hardware reports the granularity at which an invalidation request was processed through the CAIG field at the time of reporting invalidation completion (by clearing the ICC field). The following are the encoding for the 2-bit CAIG field. 00: Reserved. This is the value on reset. 01: Global Invalidation performed. The processor sets this in response to a global invalidation request. 10: Domain-selective invalidation performed using the domain-id that was specified by software in the DID field. The processor set this in response to a domain-selective or device-selective invalidation request. 11: Device-selective invalidation. The processor never sets this encoding.

Type: MEM **Bus:** 0 **Offset:** 0x28, 0x1028		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
33:32	RW	0x0	fm: Function Mask Used by the processor when performing device selective invalidation.
31:16	RW	0x0	source_id: Used by the processor when performing device selective context cache invalidation
15:0	RW	0x0	domain_id: Indicates the id of the domain whose context-entries needs to be selectively invalidated. S/W needs to program this for both domain and device selective invalidates. The processor ignores bits 15:8 since it supports only a 8-bit Domain ID.

7.5.8 vtd[0:1]_fltsts

Intel VT-d Fault Status.

Type: MEM **Bus:** 0 **Offset:** 0x34, 0x1034		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
15:8	ROS_V	0x0	fault_record_index: This field is valid only when the Primary Fault Pending field is set. This field indicates the index (from base) of the fault recording register to which the first pending fault was recorded when the Primary Fault pending field was set by hardware.
6:6	RW1CS	0x0	invalidation_timeout_error: Hardware detected a Device-IOTLB invalidation completion time-out. At this time, a fault event may be generated based on the programming of the Fault Event Control register.
5:5	RW1CS	0x0	invalidation_completion_error: Hardware received an unexpected or invalid Device-IOTLB invalidation completion. At this time, a fault event is generated based on the programming of the Fault Event Control register.
4:4	RW1CS	0x0	invalidation_queue_error: Hardware detected an error associated with the invalidation queue. For example, hardware detected an erroneous or un-supported Invalidation Descriptor in the Invalidation Queue. At this time, a fault event is generated based on the programming of the Fault Event Control register.
1:1	ROS_V	0x0	primary_fault_pending: This field indicates if there are one or more pending faults logged in the fault recording registers. Hardware computes this field as the logical OR of Fault (F) fields across all the fault recording registers of this DMA-remap hardware unit.0: No pending faults in any of the fault recording registers 1: One or more fault recording registers has pending faults. The fault recording index field is updated by hardware whenever this field is set by hardware. Also, depending on the programming of fault event control register, a fault event is generated when hardware sets this field.
0:0	RW1CS	0x0	primary_fault_overflow: Hardware sets this bit to indicate overflow of fault recording registers

7.5.9 nonisoch_fltevtctrl

Fault Event Control.

Type: MEM Bus: 0 Offset: 0x38		PortID: 8’h7e Device: 5	Function: 0
Bit	**Attr**	**Default**	**Description**
31:31	RW	0x1	fault_nonisoch_msgmsk: 1: Hardware is prohibited from issuing interrupt message requests. 0: Software has cleared this bit to indicate interrupt service is available. When a faulting condition is detected, hardware may issue a interrupt request (using the fault event data and fault event address register values) depending on the state of the interrupt mask and interrupt pending bits.
30:30	RO-V	0x0	fault_nonisoch_msi_pend: Hardware sets the IP field whenever it detects an interrupt condition. Interrupt condition is defined as when an interrupt condition occurs when hardware records a fault through one of the Fault Recording registers and sets the PPF field in Fault Status register. - Hardware detected error associated with the Invalidation Queue, setting the IQE field in the Fault Status register. - Hardware detected invalidation completion timeout error, setting the ICT field in the Fault Status register. - If any of the above status fields in the Fault Status register was already set at the time of setting any of these fields, it is not treated as a new interrupt condition. The IP field is kept set by hardware while the interrupt message is held pending. The interrupt message could be held pending due to interrupt mask (IM field) being set, or due to other transient hardware conditions. The IP field is cleared by hardware as soon as the interrupt message pending condition is serviced. This could be due to either (a) Hardware issuing the interrupt message due to either change in the transient hardware condition that caused interrupt message to be held pending or due to software clearing the IM field. (b) Software servicing all the pending interrupt status fields in the Fault Status register. - PPF field is cleared by hardware when it detects all the Fault Recording registers have Fault (F) field clear. - Other status fields in the Fault Status register is cleared by software writing back the value read from the respective fields.
29:0	RO	0x0	fault_nonisoch_msgmsk_const:

7.5.10 nonisoch_fltevtdata

Fault Event Data.

Type: MEM Bus: 0 Offset: 0x3c		PortID: 8’h7e Device: 5	Function: 0
Bit	**Attr**	**Default**	**Description**
31:16	RO	0x0	fault_nonisoch_data_const:
15:0	RW	0x0	fault_nonisoch_data:

7.5.11 vtd[0:1]_fltevtaddr

Intel VT-d Fault Event Address.

Type: **MEM** **Bus:** **0** **Offset:** **0x40, 0x1040**			**PortID:** **8'h7e** **Device:** **5** **Function:** **0**
Bit	**Attr**	**Default**	**Description**
31:2	RW	0x0	interrupt_address: The interrupt address is interpreted as the address of any other interrupt from a PCI Express* port.
1:0	RO	0x0	Reserved (Rsvd): Reserved.

7.5.12 vtd[0:1]_fltevtupraddr

Type: **MEM** **Bus:** **0** **Offset:** **0x44, 0x1044**			**PortID:** **8'h7e** **Device:** **5** **Function:** **0**
Bit	**Attr**	**Default**	**Description**
31:0	RW	0x0	address:

7.5.13 vtd[0:1]_pmen

Intel VT-d Protect Memory Enable.

Type: **MEM** **Bus:** **0** **Offset:** **0x64, 0x1064**			**PortID:** **8'h7e** **Device:** **5** **Function:** **0**
Bit	**Attr**	**Default**	**Description**
31:31	RW	0x0	protmemen: Enable Protected Memory PROT_LOW_BASE/LIMIT and PROT_HIGH_BASE/LIMIT memory regions. Software can use the protected low/high address ranges to protect both the DMA remapping tables and the interrupt remapping tables. There is no separate set of registers provided for each.
0:0	RO-V	0x0	protregionsts: This bit is set by the processor whenever it has completed enabling the protected memory region per the rules stated in the Intel VT-d specification

7.5.14 vtd[0:1]_prot_low_mem_base

Intel VT-d Protected Memory Low Base

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x68, 0x1068**				

Bit	Attr	Default	Description
31:21	RW	0x0	addr: 16MB aligned base address of the low protected DRAM region **Note:** Intel VT-d engine generated reads/writes (page walk, interrupt queue, invalidation queue read, invalidation status) themselves are allowed toward this region, but no DMA accesses (non-translated DMA or ATS translated DMA or pass through DMA, that is, no DMA access of any kind) from any device is allowed toward this region (regardless of whether TE is 0 or 1), when enabled.

7.5.15 vtd[0:1]_prot_low_mem_limit

Intel VT-d Protected Memory Low Limit.

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x6c, 0x106c**				

Bit	Attr	Default	Description
31:21	RW	0x0	addr: 16MB aligned limit address of the low protected DRAM region **Note:** The Intel VT-d engine generated reads/writes (page walk, interrupt queue, invalidation queue read, invalidation status) themselves are allowed toward this region, but no DMA accesses (non-translated DMA or ATS translated DMA or pass through DMA, that is, no DMA access of any kind) from any device is allowed toward this region (regardless of whether TE is 0 or 1), when enabled.

7.5.16 vtd[0:1]_prot_high_mem_base

Intel VT-d Protected Memory High Base.

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0x70, 0x1070**				

Bit	Attr	Default	Description
63:21	RW	0x0	addr: 16MB aligned base address of the high protected DRAM region **Note**: Intel VT-d engine generated reads/writes (page walk, interrupt queue, invalidation queue read, invalidation status) themselves are allowed toward this region, but no DMA accesses (non-translated DMA or ATS translated DMA or pass through DMA, that is, no DMA access of any kind) from any device is allowed toward this region (regardless of whether TE is 0 or 1), when enabled.

7.5.17 vtd[0:1]_prot_high_mem_limit

Intel VT-d Protected Memory High Limit.

Type: MEM **Bus:** 0 **Offset:** 0x78, 0x1078		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:21	RW	0x0	addr: 16MB aligned limit address of the high protected DRAM region **Note**: Intel VT-d engine generated reads/writes (page walk, interrupt queue, invalidation queue read, invalidation status) themselves are allowed toward this region, but no DMA accesses (non-translated DMA or ATS translated DMA or pass through DMA, that is, no DMA access of any kind) from any device is allowed toward this region (regardless of whether TE is 0 or 1), when enabled.

7.5.18 vtd[0:1]_inv_queue_head

Intel VT-d Invalidation Queue Header Pointer.

Type: MEM **Bus:** 0 **Offset:** 0x80, 0x1080		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
18:4	RO-V	0x0	queue_head: Specifies the offset (128-bit aligned) to the invalidation queue for the command that will be fetched next by hardware. This field is incremented after the command has been fetched successfully and has been verified to be a valid/supported command.

7.5.19 vtd[0:1]_inv_queue_tail

Intel VT-d Invalidation Queue Tail Pointer

Type: MEM **Bus:** 0 **Offset:** 0x88, 0x1088		**PortID:** 8'h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
18:4	RW	0x0	queue_tail: Specifies the offset (128-bit aligned) to the invalidation queue for the command that will be written next by software.

7.5.20 vtd[0:1]_inv_queue_add

Intel VT-d Invalidation Queue Address.

Type: MEM **Bus:** 0 **Offset:** 0x90, 0x1090	**PortID:** 8'h7e **Device:** 5	**Function:** 0	
Bit	**Attr**	**Default**	**Description**
63:12	RW	0x0	invreq_queue_base_address: This field points to the base of size-aligned invalidation request queue.
2:0	RW	0x0	queue_size: This field specifies the length of the invalidation request queue. The number of entries in the invalidation queue is defined as 2^(X + 8) , where X is the value programmed in this field.

7.5.21 vtd[0:1]_inv_comp_status

Intel VT-d Invalidation Completion Status.

Type: MEM **Bus:** 0 **Offset:** 0x9c, 0x109c	**PortID:** 8'h7e **Device:** 5	**Function:** 0	
Bit	**Attr**	**Default**	**Description**
0:0	RW1CS	0x0	invalidation_wait_descriptor_complete: Indicates completion of Invalidation Wait Descriptor with Interrupt Flag (IF) field set. Hardware clears this field whenever it is executing a wait descriptor with IF field set and sets this bit when the descriptor is complete.

7.5.22 nonisoch_inv_cmp_evtctrl

Invalidation Completion Event Control.

Type: MEM **Bus:** 0 **Offset:** 0xa0		**PortID:** 8’h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:31	RW	0x1	inval_nonisoch_msgmsk: 0: No masking of interrupt. When a invalidation event condition is detected, hardware issues an interrupt message (using the Invalidation Event Data and Invalidation Event Address register values). 1: This is the value on reset. Software may mask interrupt message generation by setting this field. Hardware is prohibited from sending the interrupt message when this field is set.
30:30	RO-V	0x0	inval_nonisoch_msi_pend: Hardware sets the IP field whenever it detects an interrupt condition. Interrupt condition is defined as:- An Invalidation Wait Descriptor with Interrupt Flag (IF) field set completed, setting the IWC field in the Fault Status register. - If the IWC field in the Invalidation Event Status register was already set at the time of setting this field, it is not treated as a new interrupt condition. The IP field is kept set by hardware while the interrupt message is held pending. The interrupt message could be held pending due to interrupt mask (IM field) being set, or due to other transient hardware conditions. The IP field is cleared by hardware as soon as the interrupt message pending condition is serviced. This could be due to either: (a) Hardware issuing the interrupt message due to either change in the transient hardware condition that caused interrupt message to be held pending or due to software clearing the IM field. (b) Software servicing the IWC field in the Fault Status register.
29:0	RO	0x0	inval_nonisoch_msgmsk_const:

7.5.23 nonisoch_invevtdata

Invalidation Event Data.

Type: MEM **Bus:** 0 **Offset:** 0xa4		**PortID:** 8’h7e **Device:** 5	**Function:** 0
Bit	**Attr**	**Default**	**Description**
31:16	RO	0x0	inval_nonisoch_data_const:
15:0	RW	0x0	inval_nonisoch_data:

7.5.24 vtd[0:1]_inv_comp_evt_addr

Intel VT-d Invalidation Completion Event Address.

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xa8, 0x10a8**				

Bit	Attr	Default	Description
31:2	RW	0x0	interrupt_address:
1:0	RO	0x0	reserved: 1

7.5.25 vtd[0:1]_inv_comp_evt_upraddr

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xac, 0x10ac**				

Bit	Attr	Default	Description
31:0	RW	0x0	address:

7.5.26 vtd[0:1]_intr_remap_table_base

Intel VT-d Interrupt Remapping Table Based Address.

Type:	**MEM**	**PortID:**	**8'h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**0**
Offset:	**0xb8, 0x10b8**				

Bit	Attr	Default	Description
63:12	RW	0x0	intr_remap_base: This field points to the base of page-aligned interrupt remapping table. If the Interrupt Remapping Table is larger than 4KB in size, it must be size-aligned.Reads of this field returns value that was last programmed to it.
11:11	RW-LB	0x0	ia32_extended_interrupt_enable: 0: IA-32 system is operating in legacy IA-32 interrupt mode. Hardware interprets only 8-bit APICID in the Interrupt Remapping Table entries. 1: IA-32 system is operating in extended IA-32 interrupt mode. Hardware interprets 32-bit APICID in the Interrupt Remapping Table entries.
3:0	RW	0x0	size: This field specifies the size of the interrupt remapping table. The number of entries in the interrupt remapping table is 2^(X+1), where X is the value programmed in this field.

7.5.27 vtd0_fltrec[0:7]_gpa, vtd1_fltrec0_gpa

Intel VT-d Fault Record.

Type: **MEM** **PortID:** **8’h7e**
Bus: **0** **Device:** **5** **Function:** **0**
Offset: **vtd0: 0x110, 0x120, 0x130, 0x140, 0x150, 0x160, 0x170**
vtd1: 0x1100

Bit	Attr	Default	Description
63:12	ROS_V	0x0	gpa: 4K aligned GPA for the faulting transction. valid only when F field is set.

7.5.28 vtd0_fltrec[0:7]_src, vtd1_fltrec0_src

Intel VT-d Fault Record.

Type: **MEM** **PortID:** **8’h7e**
Bus: **0** **Device:** **5** **Function:** **0**
Offset: **vtd0: 0x108, 0x118, 0x128, 0x138, 0x148, 0x158, 0x168, 0x178**
vtd1: 0x1108

Bit	Attr	Default	Description
63:63	RW1CS	0x0	f: Fault. Hardware sets this field to indicate a fault is logged in this fault recording register. The F field is set by hardware after the details of the fault is recorded in the PADDR, SID, FR and T fields.When this field is set, hardware may collapse additional faults from the same requestor (SID). Software writes the value read from this field to clear it.
62:62	ROS_V	0x0	type: Type of the first faulted DMA request 0: DMA write 1: DMA read request This field is only valid when Fault (F) bit is set.
61:60	ROS_V	0x0	address_type: This field captures the AT field from the faulted DMA request. This field is valid only when the F field is set.
39:32	ROS_V	0x0	fault_reason: Reason for the first translation fault. See Intel VT-d specification for details.This field is only valid when Fault bit is set.
15:0	ROS_V	0x0	source_identifier: Requester ID of the dma request that faulted. Valid only when F bit is set

7.5.29 vtd[0:1]_invaddrreg

Intel VT-d Invalidate Address.

Type: MEM **Bus:** 0 **Offset:** 0x200, 0x1200	**PortID:** 8'h7e **Device:** 5		**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:12	RW	0x0	addr: To request a page-specific invalidation request to hardware, software must first write the corresponding guest physical address to this register, and then issue a page-specific invalidate command through the IOTLB_REG.
6:6	RW	0x0	ih: The field provides hint to hardware to preserve or flush the respective non-leaf page-table entries that may be cached in hardware.0: Software may have modified both leaf and non-leaf page-table entries corresponding to mappings specified in the ADDR and AM fields. On a page-selective invalidation request, IIO must flush both the cached leaf and nonleaf page-table entries corresponding to mappings specified by ADDR and AM fields. IIO performs a domain-level invalidation on non-leaf entries and page-selective-domain-level invalidation at the leaf level 1: Software has not modified any non-leaf page-table entries corresponding to mappings specified in the ADDR and AM fields. On a page-selective invalidation request, IIO preserves the cached non-leaf page-table entries corresponding to mappings specified by ADDR and AM fields and performs only a page-selective invalidation at the leaf level
5:0	RW	0x0	am: IIO supports values of 0-9. All other values result in undefined results.

7.5.30 vtd[0:1]_iotlbinv

Intel VT-d IOTLB Invalidate.

Type: MEM **Bus:** 0 **Offset:** 0x208, 0x1208	**PortID:** 8'h7e **Device:** 5		**Function:** 0
Bit	**Attr**	**Default**	**Description**
63:63	RW-V	0x0	Intel VT: Invalidate IOTLB cache Software requests IOTLB invalidation by setting this field. Software must also set the requested invalidation granularity by programming the IIRG field.Hardware clears the Intel VT field to indicate the invalidation request is complete. Hardware also indicates the granularity at which the invalidation operation was performed through the IAIG field. Software must read back and check the processor field to be clear to confirm the invalidation is complete. When processor field is set, software must not update the contents of this register (and Invalidate Address register, if it is being used), nor submit new IOTLB invalidation requests.
62:62	RO	0x0	rsz2: Reserved.

Type: **MEM** **Bus:** **0** **Offset:** **0x208, 0x1208**		**PortID:** **8’h7e** **Device:** **5**	**Function:** **0**

Bit	Attr	Default	Description
61:60	RW	0x0	iirg: IOTLB Invalidation Request Granularity When requesting hardware to invalidate the I/OTLB (by setting the Intel VT field), software writes the requested invalidation granularity through this IIRG field. Following are the encoding for the 2-bit IIRG field. 00: Reserved. Hardware ignores the invalidation request and reports invalidation complete by clearing the Intel VT field and reporting 00 in the AIG field. 01: Global Invalidation request. 10: Domain-selective invalidation request. The target domain-id must be specified in the DID field. 11: Page-selective invalidation request. The target address, mask and invalidation hint must be specified in the Invalidate Address register, the domain-id must be provided in the DID field.
59:59	RO	0x0	rsz1: Reserved.
58:57	RO-V	0x0	iaig: IOTLB Actual Invalidation Granularity Hardware reports the granularity at which an invalidation request was proceed through the AIG field at the time of reporting invalidation completion (by clearing the Intel VT field).The following are the encoding for the 2-bit IAIG field. 00: Reserved. This indicates hardware detected an incorrect invalidation request and ignored the request. Examples of incorrect invalidation requests include detecting an unsupported address mask value in Invalidate Address register for page-selective invalidation requests or an unsupported/undefined encoding in IIRG. 01: Global Invalidation performed. The processor sets this in response to a global IOTLB invalidation request. 10: Domain-selective invalidation performed using the domain-id that was specified by software in the DID field. The processor sets this in response to a domain selective IOTLB invalidation request. 11: Processor sets this in response to a page selective invalidation request.
49:49	RW	0x0	dr: Processor uses this to drain or not drain reads on an invalidation request.
48:48	RW	0x0	dw: Processor uses this to drain or not drain reads on an invalidation request.
47:32	RW	0x0	did: Domain to be invalidated and is programmed by software for both page and domain selective invalidation requests. processor ignores the bits 47:40 since it supports only an 8-bit Domain ID.

7.6 Device 5 Function 1

7.6.1 vid

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 1
Offset: 0x0

Bit	Attr	Default	Description
15:0	RO	0x8086	vendor_identification_number: The value is assigned by PCI-SIG to Intel.

7.6.2 did

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 1
Offset: 0x2

Bit	Attr	Default	Description
15:0	RO	0xe29	device_identification_number: Device ID values vary from function to function. Bits 15:8 are equal to 0x0E .

7.6.3 pcicmd

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 1
Offset: 0x4

Bit	Attr	Default	Description
10:10	RW	0x0	intx_interrupt_disable: 1

7.6.4 pcists

Type: CFG **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 1
Offset: 0x6

Bit	Attr	Default	Description
4:4	RO	0x1	capl: 1
3:3	RO-V	0x1	intxstat: 1

7.6.5 rid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x8**				

Bit	Attr	Default	Description
7:0	RO-V	0x0	revision_id: Reflects the Uncore Revision ID after reset. Reflects the Compatibility Revision ID after the BIOS writes 0x69 to any RID register in any processor function. **Implementation Note:** Read and write requests from the host to any RID register in any processor function are re-directed to the IIO cluster. Accesses to the CCR field are also redirected due to Dword alignment. It is possible that JTAG accesses are direct, so will not always be redirected.

7.6.6 ccr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x9**				

Bit	Attr	Default	Description
23:16	RO	0x8	base_class: Generic Device
15:8	RO	0x80	sub_class: Generic Device
7:0	RO	0x0	interface: 1

7.6.7 clsr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0xc**				

Bit	Attr	Default	Description
7:0	RW	0x0	cacheline_size: This register is set as RW for compatibility reasons only. Cacheline size is always 64B.

7.6.8 plat

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0xd**				

Bit	Attr	Default	Description
7:0	RO	0x0	primary_latency_timer: Not applicable to PCI-Express*. Hardwired to 00h.

7.6.9 hdr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0xe**				

Bit	Attr	Default	Description
7:7	RO	0x0	multi_function_device: This bit defaults to 1b since all these devices are multi-function
6:0	RO	0x0	configuration_layout: This field identifies the format of the configuration header layout. It is Type 0 for all these devices. The default is 00h, indicating a 'endpoint device'.

7.6.10 bist

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0xf**				

Bit	Attr	Default	Description
7:0	RO	0x0	bist_tests: Not supported. Hardwired to 00h

7.6.11 svid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x2c**				

Bit	Attr	Default	Description
15:0	RW-O	0x0	subsystem_vendor_identification_number: The default value specifies Intel but can be set to any value once after reset.

7.6.12 sdid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x2e**				

Bit	Attr	Default	Description
15:0	RW-O	0x0	subsystem_device_identification_number: Assigned by the subsystem vendor to uniquely identify the subsystem

7.6.13 capptr

Type: CFG **Bus: 0** **Offset: 0x34**	**PortID: N/A** **Device: 5**	**Function: 1**	
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x40	capability_pointer: Points to the first capability structure for the device which is the PCIe capability.

7.6.14 intl

Type: CFG **Bus: 0** **Offset: 0x3c**	**PortID: N/A** **Device: 5**	**Function: 1**	
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x0	interrupt_line: N/A for these devices

7.6.15 intpin

Type: CFG **Bus: 0** **Offset: 0x3d**	**PortID: N/A** **Device: 5**	**Function: 1**	
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x0	interrupt_pin: N/A since these devices do not generate any interrupt on their own

7.6.16 mingnt

Type: CFG **Bus: 0** **Offset: 0x3e**	**PortID: N/A** **Device: 5**	**Function: 1**	
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x0	mgv:

7.6.17 maxlat

Type: CFG **Bus: 0** **Offset: 0x3f**	**PortID: N/A** **Device: 5**	**Function: 1**	
Bit	**Attr**	**Default**	**Description**
7:0	RO	0x0	mlv:

7.6.18 pxpcap

Type: CFG Bus: 0 Offset: 0x40e	PortID: N/A Device: 5	Function: 1	
Bit	**Attr**	**Default**	**Description**
29:25	RO	0x0	interrupt_message_number: N/A for this device
24:24	RO	0x0	slot_implemented: N/A for integrated endpoints
23:20	RO	0x9	device_port_type: Device type is Root Complex Integrated Endpoint
19:16	RO	0x1	capability_version: PCI Express* Capability is Compliant with Version 1.0 of the PCI Express* Specification. **Note:** This capability structure is not compliant with Versions beyond 1.0, since they require additional capability registers to be reserved. The only purpose for this capability structure is to make enhanced configuration space available. Minimizing the size of this structure is accomplished by reporting version 1.0 compliancy and reporting that this is an integrated root port device. As such, only three Dwords of configuration space are required for this structure.
15:8	RO	0x80	next_ptr: Pointer to the next capability. Set to 0 to indicate there are no more capability structures.
7:0	RO	0x10	capability_id: Provides the PCI Express* capability ID assigned by PCI-SIG.

7.6.19 msicap

MSI Capability.

Type: CFG Bus: 0 Offset: 0x80	PortID: N/A Device: 5	Function: 1	
Bit	**Attr**	**Default**	**Description**
15:8	RO	0x0	next_ptr: Next pointer. 0: There are no other capability structures in the lower config space
7:0	RO	0x5	capability_id: 05 for MSI capability.

7.6.20 msictl

MSI Control.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x82**				

Bit	Attr	Default	Description
15:9	RV	0x0	Reserved
8:8	RO	0x0	pvmc: Per Vector Masking Capable. This function does not support per vector masking.
7:7	RO	0x1	b64ac: 64-bit Address Capable. This function is 64-bit address capable.
6:4	RO	0x0	mme: Multiple Message Enable. This function only supports one vector.
3:1	RO	0x0	mmc: Multiple Message Capable. This function only requests one vector.
0:0	RW	0x0	msien: MSI Enable. Enables MSI's from this function if set. If cleared, then this function will generate legacy interrupts.

7.6.21 msiar

The MSI Address Register MSIAR contains the system specific address information to route MSI interrupts from the root ports and is broken into its constituent fields.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x84**				

Bit	Attr	Default	Description
63:2	RW	0x0	msi_address: MSI Address. (Dword aligned)
1:0	RV	0x0	Reserved

7.6.22 msidr

MSI Data.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x8c**				

Bit	Attr	Default	Description
15:0	RW	0x0	msidr_data: Message Data.

7.6.23 memhpctrl

Memory Hot-Plug Control.

Type: CFG **Bus:** 0 **Offset:** 0xa0	**PortID:** N/A **Device:** 5	**Function:** 1	
Bit	**Attr**	**Default**	**Description**
31:1	RV	0x0	Reserved
0:0	RW	0x0	smien: SMI Enable. Enable SMI interrupt generation on any hot-plug event (regardless of whether it is enabled in the MemHP capabilities).

7.6.24 xpprivc1

Type: CFG **Bus:** 0 **Offset:** 0xd0	**PortID:** N/A **Device:** 5	**Function:** 1	
Bit	**Attr**	**Default**	**Description**
5:5	RWS	0x0	hpmsiclapsen: 1
4:4	RWS	0x1	hpmsirevalen:

7.6.25 memhpcap[0:3]

Channel X Memory Hot-Plug Capability (X = 0, 1, 2, 3)

Type: CFG **Bus:** 0 **Offset:** 0x100, 0x110, 0x120, 0x130	**PortID:** N/A **Device:** 5	**Function:** 1	
Bit	**Attr**	**Default**	**Description**
31:20	RO	0x110 (memhpcap0) 0x120 (memhpcap1) 0x130 (memhpcap2) 0x0 (memhpcap3)	next_ptr: Next Pointer. This points to the next capability structure.
19:16	RO	0x1	capability_version:
15:0	RO	0xb	vendor_specific_capability:

7.6.26 memhphdr[0:3]

Channel X Memory Hot-Plug Capability Header. (X = 0, 1, 2, 3)

Type: CFG **Bus:** 0 **Offset:** 0x104, 0x114, 0x124, x134	**PortID:** N/A **Device:** 5	**Function:** 1	
Bit	**Attr**	**Default**	**Description**
31:20	RO	0x10	vendor_specific_length: There are 16 bytes in this capability structure.
19:16	RO	0x1	vendor_specific_revision_id: First revision of this capability structure.
15:0	RO	0x6	vendor_specific_id: Represents the Memory Hot-Plug Capability.

7.6.27 sltcap[0:3]

Channel X Slot Capability (X=0, 1, 2, 3)

Type: CFG **Bus:** 0 **Offset:** 0x108, 0x118, 0x128, 0x138	**PortID:** N/A **Device:** 5	**Function:** 1	
Bit	**Attr**	**Default**	**Description**
31:19	RW-O	0x0	physical_slot_number: Indicates the associated memory channel number.
18:18	RO	0x0	command_complete_not_capable: If set, indicates that this structure is not capable of generating an interrupt on completion of the last command.
17:17	RW-O	0x0	electromechanical_interlock_present: This bit when set indicates that an Electromechanical Interlock is implemented on the chassis for this slot and that lock is controlled by bit 11 in Slot Control register. This field is initialized by the BIOS based on the system architecture. **BIOS Note:** This capability is not set if the Electromechanical Interlock control is connected to main slot power control. This is expected to be used only for hot-pluggable slots.
16:7	RV	0x0	Reserved
6:6	RW-O	0x0	hot_plug_capable: This field defines hot-plug support capabilities for the Memory Channel 0: indicates that this slot is not capable of supporting Hot-plug operations. 1: indicates that this slot is capable of supporting Hot-plug operations This bit is programmed by the BIOS based on the system design. This bit must be programmed by the BIOS to be consistent with the VPP enable bit for the port.
5:5	RO	0x0	hot_plug_surprise: This field indicates that a device in this slot may be removed from the system without prior notification. This field is initialized by the BIOS. 0: indicates that hot-plug surprise is not supported 1: indicates that hot-plug surprise is supported This bit is not set because there are no known usage models and no hardware mechanism for detecting a surprise hot-plug event.

Type: CFG Bus: 0 Offset: 0x108, 0x118, 0x128, 0x138		PortID: N/A Device: 5	Function: 1
Bit	**Attr**	**Default**	**Description**
4:4	RW-O	0x0	power_indicator_present: This bit indicates that a Power Indicator is implemented for this slot and is electrically controlled by the chassis. 0: indicates that a Power Indicator that is electrically controlled by the chassis is not present 1: indicates that Power Indicator that is electrically controlled by the chassis is present BIOS programs this field.
3:3	RW-O	0x0	attention_indicator_present: This bit indicates that an Attention Indicator is implemented for this slot and is electrically controlled by the chassis 0: indicates that an Attention Indicator that is electrically controlled by the chassis is not present 1: indicates that an Attention Indicator that is electrically controlled by the chassis is present BIOS programs this field.
2:2	RW-O	0x0	mrl_sensor_present: This bit indicates that an MRL Sensor is implemented on the chassis for this slot. 0: indicates that an MRL Sensor is not present 1: indicates that an MRL Sensor is present BIOS programs this field.
1:1	RW-O	0x0	power_controller_present: This bit indicates that a software controllable power controller is implemented on the chassis for this slot. 0: indicates that a software controllable power controller is not present 1: indicates that a software controllable power controller is present BIOS programs this field.
0:0	RW-O	0x0	attention_button_present: This bit indicates that the Attention Button event signal is routed (from slot or on-board in the chassis) to the IIO's hot-plug controller. 0: indicates that an Attention Button signal is routed to IIO 1: indicates that an Attention Button is not routed to IIO BIOS programs this field.

7.6.28 sltcon[0:3]

Channel X Slot Control (X=0, 1, 2, 3)

Type:	**CFG**	**PortID:**	**N/A**
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x10c, 0x11c, 0x12c, 0x13c**		

Bit	Attr	Default	Description
15:12	RV	0x0	Reserved
11:11	RWS	0x0	electromechanical_interlock_control: When software writes a 1 to this bit, IIO pulses the EMIL pin. Write of 0 has no effect. This bit always returns a 0 when read. If electromechanical lock is not implemented, then either a write of 1 or 0 to this register has no effect.
10:10	RWS	0x1	power_controller_control: If a power controller is implemented, when writes to this field will set the power state of the slot as indicated by this bit. Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 0: Power On 1: Power Off
9:8	RW	0x3	power_indicator_control: If a Power Indicator is implemented, writes to this field will set the Power Indicator to the written state. Reads of this field must reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 00: Reserved 01: On 10: Blink (IIO drives 1 Hz square wave for Chassis mounted LEDs) 11: Off
7:6	RW	0x3	attention_indicator_control: If an Attention Indicator is implemented, writes to this field will set the Attention Indicator to the written state. Reads of this field reflect the value from the latest write, even if the corresponding hot-plug command is not executed yet at the VPP, unless software issues a write without waiting for the previous command to complete in which case the read value is undefined. 00: Reserved 01: On 10: Blink (IIO drives 1 Hz square wave) 11: Off
5:5	RW	0x0	hot_plug_interrupt_enable: When set to 1b, this bit enables generation of Hot-Plug interrupt, MSI or INTx interrupt depending on the setting of the MSI enable bit in 'MSI Control Register (MSICTRL)' on enabled Hot-Plug events. 0: Disables interrupt generation on Hot-plug events 1: Enables interrupt generation on Hot-plug events
4:4	RW	0x0	command_completed_interrupt_enable: This field enables software notification (Interrupt - MSI/INTx) when a command is completed by the Hot-plug controller connected to the PCI Express* port 0: Disables hot-plug interrupts on a command completion by a hot-plug Controller 1: Enables hot-plug interrupts on a command completion by a hot-plug Controller

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x10c, 0x11c, 0x12c, 0x13c**				

Bit	Attr	Default	Description
3:3	RW	0x0	presence_detect_changed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a presence detect changed event. 0: Disables generation of hot-plug interrupts when a presence detect changed event happens. 1: Enables generation of hot-plug interrupts when a presence detect changed event happens.
2:2	RW	0x0	mrl_sensor_changed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a MRL Sensor changed event. 0: Disables generation of hot-plug interrupts when an MRL Sensor changed event happens. 1: Enables generation of hot-plug interrupts when an MRL Sensor changed event happens.
1:1	RW	0x0	power_fault_detected_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of a power fault event. 0: Disables generation of hot-plug interrupts when a power fault event happens. 1: Enables generation of hot-plug interrupts when a power fault event happens.
0:0	RW	0x0	attention_button_pressed_enable: This bit enables the generation of hot-plug interrupts or wake messages by means of an attention button pressed event. 0: Disables generation of hot-plug interrupts when the attention button is pressed. 1: Enables generation of hot-plug interrupts when the attention button is pressed.

7.6.29 sltsts[0:3]

Channel X Slot Status. (X=0, 1, 2, 3)

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x10e, 0x11e, 0x12e, 0x13e**				

Bit	Attr	Default	Description
15:8	RV	0x0	Reserved
7:7	RO	0x0	electromechanical_latch_status: When read this register returns the current state of the Electromechanical Interlock (the EMILS pin) which has the defined encodings as: 0: Electromechanical Interlock Disengaged 1: Electromechanical Interlock Engaged
6:6	RO	0x0	presence_detect_state: When read, this register returns the current state of the Present Detect pin. 0: Module slot empty 1: Module Present in slot (powered or unpowered)
5:5	RO	0x0	mrl_sensor_state: This bit reports the status of an MRL sensor if it is implemented. 0: MRL Closed 1: MRL Open

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**1**
Offset:	**0x10e, 0x11e, 0x12e, 0x13e**				

Bit	Attr	Default	Description
4:4	RW1C	0x0	command_completed: This bit is set by IIO when the hot-plug command has completed and the hot-plug controller is ready to accept a subsequent command. It is subsequently cleared by software after the field has been read and processed. This bit provides no guarantee that the action corresponding to the command is complete. Any write to SLTCON (regardless of the port is capable or enabled for hot-plug) is considered a 'hot-plug' command. If the port is not hot-plug capable or hot-plug enabled, then the hot-plug command does not trigger any action on the VPP port but the command is still completed by means of this bit.
3:3	RW1C	0x0	presence_detect_changed: This bit is set by IIO when the value reported in bit 6 is changes. It is subsequently cleared by software after the field has been read and processed.
2:2	RW1C	0x0	mrl_sensor_changed: This bit is set if the value reported in bit 5 changes. It is subsequently cleared by software after the field has been read and processed.
1:1	RW1C	0x0	power_fault_detected: This bit is set by IIO when a power fault event is detected by the power controller (which is reported by means of the VPP bit stream). It is subsequently cleared by software after the field has been read and processed.
0:0	RW1C	0x0	attention_button_pressed: This bit is set by IIO when the attention button is pressed. It is subsequently cleared by software after the field has been read and processed.

7.7 Device 5 Function 2

Global System Control and Error Registers.

Table 7-9. Integrated I/O Device 5 Function 2 Register Address Map (Sheet 1 of 3)

Register Name	Offset	Size
vid	0x0	16
did	0x2	16
pcicmd	0x4	16
pcists	0x6	16
rid	0x8	8
ccr	0x9	24
clsr	0xc	8
hdr	0xe	8
svid	0x2c	16
sdid	0x2e	16
capptr	0x34	8
intl	0x3c	8
intpin	0x3d	8
pxpcapid	0x40	8
pxpnxtptr	0x41	8
pxpcap	0x42	16
csr_sat_mask_set	0x46	16
cgctrl3	0x48	32
cgctrl6	0x4c	32
cgctrl7	0x50	32
cgsts	0x54	32
cgstagger	0x58	8
cgctrl5	0x59	8
cgctrl4_0	0x5a	16
cgctrl4_1	0x5c	16
irpperrsv	0x80	64
iioerrsv	0x8c	32
mierrsv	0x90	32
pcierrsv	0x94	32
sysmap	0x9c	32
viral	0xa0	32
errpinctl	0xa4	32
errpinsts	0xa8	32
errpindat	0xac	32
vppctl	0xb0	64
vppsts	0xb8	32
vppfreq	0xbc	32

Table 7-9. Integrated I/O Device 5 Function 2 Register Address Map (Sheet 2 of 3)

Register Name	Offset	Size
vppmem	0xc0	64
vpp_inverts	0xc8	32
miscprivc	0x16c	32
gcerrst	0x1a8	32
gcferrst	0x1ac	32
gcnerrst	0x1b8	32
gnerrst	0x1c0	32
gferrst	0x1c4	32
gerrctl	0x1c8	32
gsysst	0x1cc	32
gsysctl	0x1d0	32
gfferrst	0x1dc	32
gfnerrst	0x1e8	32
gnferrst	0x1ec	32
gnnerrst	0x1f8	32
irpp0errst	0x230	32
irpp0errctl	0x234	32
irpp0fferrst	0x238	32
irpp0fnerrst	0x23c	32
irpp0fferrhd0	0x240	32
irpp0fferrhd1	0x244	32
irpp0fferrhd2	0x248	32
irpp0fferrhd3	0x24c	32
irpp0nferrst	0x250	32
irpp0nnerrst	0x254	32
irpp0nferrhd0	0x258	32
irpp0nferrhd1	0x25c	32
irpp0nferrhd2	0x260	32
irpp0nferrhd3	0x264	32
irpp0errcntsel	0x268	32
irpp0errcnt	0x26c	32
irpp1errst	0x2b0	32
irpp1errctl	0x2b4	32
irpp1fferrst	0x2b8	32
irpp1fnerrst	0x2bc	32
irpp1fferrhd0	0x2c0	32
irpp1fferrhd1	0x2c4	32
irpp1fferrhd2	0x2c8	32
irpp1fferrhd3	0x2cc	32
irpp1nferrst	0x2d0	32

Table 7-9. Integrated I/O Device 5 Function 2 Register Address Map (Sheet 3 of 3)

Register Name	Offset	Size
irpp1nnerrst	0x2d4	32
irpp1nferrhd0	0x2d8	32
irpp1nferrhd1	0x2dc	32
irpp1nferrhd2	0x2e0	32
irpp1nferrhd3	0x2e4	32
irpp1errcntsel	0x2e8	32
irpp1errcnt	0x2ec	32
iioerrst	0x300	32
iioerrctl	0x304	32
iiofferrst	0x308	32
iiofferrhd_0	0x30c	32
iiofferrhd_1	0x310	32
iiofferrhd_2	0x314	32
iiofferrhd_3	0x318	32
iiofnerrst	0x31c	32
iionferrst	0x320	32
iionferrhd_0	0x324	32
iionferrhd_1	0x328	32
iionferrhd_2	0x32c	32
iionferrhd_3	0x330	32
iionnerrst	0x334	32
iioerrcntsel	0x33c	32
iioerrcnt	0x340	32
mierrst	0x380	32
mierrctl	0x384	32
mifferrst	0x388	32
mifferrhdr_0	0x38c	32
mifferrhdr_1	0x390	32
mifferrhdr_2	0x394	32
mifferrhdr_3	0x398	32
mifnerrst	0x39c	32
minferrst	0x3a0	32
minferrhdr_0	0x3a4	32
minferrhdr_1	0x3a8	32
minferrhdr_2	0x3ac	32
minferrhdr_3	0x3b0	32
minnerrst	0x3b4	32
mierrcntsel	0x3bc	32
mierrcnt	0x3c0	8

7.7.1 vid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x0**				

Bit	Attr	Default	Description
15:0	RO	0x8086	vendor_identification_number: The value is assigned by PCI-SIG to Intel.

7.7.2 did

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x2**				

Bit	Attr	Default	Description
15:0	RO	0xe2a	device_identification_number: Device ID values vary from function to function. Bits 15:8 are equal to 0x0E.

7.7.3 pcicmd

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x4**				

Bit	Attr	Default	Description
10:10	RO	0x0	intx_disable: N/A for these devices
9:9	RO	0x0	fast_back_to_back_enable: Not applicable to PCI Express* and is hardwired to 0
8:8	RO	0x0	serr_enable: This bit has no impact on error reporting from these devices
7:7	RO	0x0	idsel_stepping_wait_cycle_control: Not applicable to internal devices. Hardwired to 0.
6:6	RO	0x0	parity_error_response: This bit has no impact on error reporting from these devices
5:5	RO	0x0	vga_palette_snoop_enable: Not applicable to internal devices. Hardwired to 0.
4:4	RO	0x0	memory_write_and_invalidate_enable: Not applicable to internal devices. Hardwired to 0.
3:3	RO	0x0	special_cycle_enable: Not applicable. Hardwired to 0.
2:2	RO	0x0	bus_master_enable: Hardwired to 0 since these devices don't generate any transactions
1:1	RO	0x0	memory_space_enable: Hardwired to 0 since these devices don't decode any memory BARs
0:0	RO	0x0	io_space_enable: Hardwired to 0 since these devices don't decode any I/O BARs

7.7.4 pcists

Type: CFG Bus: 0 Offset: 0x6	PortID: N/A Device: 5	Function: 2	
Bit	**Attr**	**Default**	**Description**
15:15	RO	0x0	detected_parity_error: This bit is set when the device receives a packet on the primary side with an uncorrectable data error including a packet with poison bit set or an uncorrectable addresscontrol parity error. The setting of this bit is regardless of the Parity Error Response bit PERRE in the PCICMD register. R2PCIe will never set this bit.
14:14	RO	0x0	signaled_system_error: Hardwired to 0
13:13	RO	0x0	received_master_abort: Hardwired to 0
12:12	RO	0x0	received_target_abort: Hardwired to 0
11:11	RO	0x0	signaled_target_abort: Hardwired to 0
10:9	RO	0x0	devsel_timing: Not applicable to PCI Express*. Hardwired to 0.
8:8	RO	0x0	master_data_parity_error: Hardwired to 0
7:7	RO	0x0	fast_back_to_back: Not applicable to PCI Express*. Hardwired to 0.
5:5	RO	0x0	pci66mhz_capable: Not applicable to PCI Express*. Hardwired to 0.
4:4	RO	0x1	capabilities_list: This bit indicates the presence of a capabilities list structure
3:3	RO	0x0	intx_status: Hardwired to 0

7.7.5 rid

Type: CFG Bus: 0 Offset: 0x8	PortID: N/A Device: 5	Function: 2	
Bit	**Attr**	**Default**	**Description**
7:0	RO-V	0x0	revision_id: Reflects the Uncore Revision ID after reset. Reflects the Compatibility Revision ID after the BIOS writes 0x69 to any RID register in any processor function. **Implementation Note:** Read and write requests from the host to any RID register in any processor function are re-directed to the IIO cluster. Accesses to the CCR field are also redirected due to Dword alignment. It is possible that JTAG accesses are direct, so will not always be redirected.

7.7.6 ccr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x9**				

Bit	Attr	Default	Description
23:16	RO-V	0x8	base_class: Generic Device
15:8	RO-V	0x80	sub_class: Generic Device
7:0	RO-V	0x0	register_level_programming_interface: Set to 00h for all non-APIC devices.

7.7.7 clsr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xc**				

Bit	Attr	Default	Description
7:0	RW	0x0	cacheline_size: This register is set as RW for compatibility reasons only. Cacheline size is always 64B.

7.7.8 hdr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xe**				

Bit	Attr	Default	Description
7:7	RO	0x1	multi_function_device: This bit defaults to 1b since all these devices are multi-function
6:0	RO	0x0	configuration_layout: This field identifies the format of the configuration header layout. It is Type 0 for all these devices. The default is 00h, indicating a 'endpoint device'.

7.7.9 svid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x2c**				

Bit	Attr	Default	Description
15:0	RW-O	0x0	subsystem_vendor_identification_number: The default value specifies Intel but can be set to any value once after reset.

7.7.10 sdid

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	2
Offset:	0x2e				

Bit	Attr	Default	Description
15:0	RW-O	0x0	subsystem_device_identification_number: Assigned by the subsystem vendor to uniquely identify the subsystem

7.7.11 capptr

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	2
Offset:	0x34				

Bit	Attr	Default	Description
7:0	RO	0x40	capability_pointer: Points to the first capability structure for the device which is the PCIe capability.

7.7.12 intl

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	2
Offset:	0x3c				

Bit	Attr	Default	Description
7:0	RO	0x0	interrupt_line: N/A for these devices

7.7.13 intpin

Type:	CFG	PortID:	N/A		
Bus:	0	Device:	5	Function:	2
Offset:	0x3d				

Bit	Attr	Default	Description
7:0	RO	0x0	interrupt_pin: N/A since these devices do not generate any interrupt on their own

7.7.14 pxpcapid

Type:	**CFG**	**PortID:**	**N/A**
Bus:	**0**	**Device:**	**5**
Offset:	**0x40**	**Function:**	**2**

Bit	Attr	Default	Description
7:0	RO	0x10	capability_id: Provides the PCI Express* capability ID assigned by PCI-SIG.

7.7.15 pxpnxtptr

Type:	**CFG**	**PortID:**	**N/A**
Bus:	**0**	**Device:**	**5**
Offset:	**0x41**	**Function:**	**2**

Bit	Attr	Default	Description
7:0	RO	0x0	next_ptr: This field is set to the PCI PM capability.

7.7.16 pxpcap

Type:	**CFG**	**PortID:**	**N/A**
Bus:	**0**	**Device:**	**5**
Offset:	**0x42**	**Function:**	**2**

Bit	Attr	Default	Description
13:9	RO	0x0	interrupt_message_number_n_a:
8:8	RO	0x0	slot_implemented_n_a:
7:4	RO	0x9	device_port_type: This field identifies the type of device. It is set to for the DMA to indicate root complex integrated endpoint device.
3:0	RO	0x2	capability_version: This field identifies the version of the PCI Express* capability structure. Set to 2h for PCI Express* and DMA devices for compliance with the extended base registers.

7.7.17 csr_sat_mask_set

Type:	**CFG**	**PortID:**	**N/A**
Bus:	**0**	**Device:**	**5**
Offset:	**0x46**	**Function:**	**2**

Bit	Attr	Default	Description
15:0	RW	0x0	csr_sat_mask_set:

7.7.18 cgctrl3

Type: CFG Bus: 0 Offset: 0x48		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:0	RW	0x0	alarm:

7.7.19 cgctrl6

Type: CFG Bus: 0 Offset: 0x4c		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:28	RW	0x0	progseq07:
27:24	RW	0x0	progseq06:
23:20	RW	0x0	progseq05:
19:16	RW	0x0	progseq04:
15:12	RW	0x0	progseq03:
11:8	RW	0x0	progseq02:
7:4	RW	0x0	progseq01:
3:0	RW	0x0	progseq00:

7.7.20 cgctrl7

Type: CFG Bus: 0 Offset: 0x50		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:28	RW	0x0	progseq15:
27:24	RW	0x0	progseq14:
23:20	RW	0x0	progseq13:
19:16	RW	0x0	progseq12:
15:12	RW	0x0	progseq11:
11:8	RW	0x0	progseq10:
7:4	RW	0x0	progseq09:
3:0	RW	0x0	progseq08:

7.7.21 cgsts

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x54**				

Bit	Attr	Default	Description
31:0	RW-V	0x0	gated_duration:

7.7.22 cgstagger

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x58**				

Bit	Attr	Default	Description
7:0	RW	0x0	stagger:

7.7.23 cgctrl5

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x59**				

Bit	Attr	Default	Description
3:0	RW	0x0	numsattelites:

7.7.24 cgctrl4_0

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x5a**				

Bit	Attr	Default	Description
15:0	RW	0x0	pstatedelay1:

7.7.25 cgctrl4_1

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x5c**				

Bit	Attr	Default	Description
2:0	RW	0x0	pstatedelay2:

7.7.26 irpperrsv

IRP Protocol Error Severity.

Type: CFG **Bus:** 0 **Offset:** 0x80		**PortID:** N/A **Device:** 5	**Function:** 2

Bit	Attr	Default	Description
29:28	RWS	0x2	protocol_parity_error: (DB) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
27:26	RWS	0x2	protocol_qt_overflow_underflow: (DA) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
21:20	RWS	0x2	protocol_rcvd_unexprsp: (D7) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
9:8	RWS	0x1	csr_acc_32b_unaligned: (C3) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
7:6	RWS	0x1	wrcache_uncecc_error: (C2) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
5:4	RWS	0x1	protocol_rcvd_poison: (C1) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved
3:2	RWS	0x0	wrcache_correcc_error: (B4) 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved

7.7.27 iioerrsv

IIO Core Error Severity.

This register associates the detected IIO internal core errors to an error severity level. An individual error is reported with the corresponding severity in this register. Software can program the error severity to one of the three severities supported by IIO. This register is sticky and can only be reset by PWRGOOD.

Type: CFG **Bus:** 0 **Offset:** 0x8c		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
13:12	RWS_L	0x1	c6_overflow_underflow_error: 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved **Note:** Locked by RSPLCK
11:10	RWS_L	0x1	RSVD
9:8	RWS_L	0x1	c4_master_abort_address_error: 00: Error Severity Level 0 (Correctable) 01: Error Severity Level 1 (Recoverable) 10: Error Severity Level 2 (Fatal) 11: Reserved **Note:** Locked by RSPLCK
7:0	RWS_L	0x0	Reserved

7.7.28 mierrsv

Miscellaneous Error Severity

Type: CFG **Bus:** 0 **Offset:** 0x90		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
9:8	RWS	0x0	RSVD
7:6	RWS	0x0	vpp_err_sts: This bit should be programmed to 1.
5:4	RWS	0x0	RSVD
3:2	RWS	0x0	RSVD
1:0	RWS	0x0	RSVD

7.7.29 pcierrsv

PCIe Error Severity Map.

This register allows remapping of the PCI-E errors to the IIO error severity.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x94**				

Bit	Attr	Default	Description
5:4	RWS	0x2	pciefaterr_map: 10: Map this PCI-E error type to Error Severity 2 01: Map this PCI-E error type to Error Severity 1 00: Map this PCI-E error type to Error Severity 0
3:2	RWS	0x1	pcienonfaterr_map: 10: Map this PCI-E error type to Error Severity 2 01: Map this PCI-E error type to Error Severity 1 00: Map this PCI-E error type to Error Severity 0
1:0	RWS	0x0	pciecorerr_map: 10: Map this PCI-E error type to Error Severity 2 01: Map this PCI-E error type to Error Severity 1 00: Map this PCI-E error type to Error Severity 0

7.7.30 sysmap

System Error Event map.

This register maps the error severity detected by the IIO to on of the system events. When an error is detected by the IIO, its corresponding error severity determines which system event to generate according to this register.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x9c**				

Bit	Attr	Default	Description
10:8	RWS	0x1	sev2_map: 010: Generate NMI 001: Generate SMIPMI 000: No inband message Others: Reserved
6:4	RWS	0x2	sev1_map: 010: Generate NMI 001: Generate SMIPMI 000: No inband message Others: Reserved
2:0	RWS	0x0	sev0_map: 010: Generate NMI 001: Generate SMIPMI 000: No inband message Others: Reserved

7.7.31 viral

This register provides the option to generate viral alert upon the detection of fatal error.

Type: CFG **Bus:** 0 **Offset:** 0xa0		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
31:31	RW1C	0x0	iio_viral_state: Indicates the IIO cluster is in a viral state. When set, all outbound requests are master aborted, all inbound requests are master aborted. This includes traffic to and from the DMI port, except the Reset_Warn message, which will be auto-completed by the DMI port. This state bit is cleared by warm reset.
30:30	RW1CS	0x0	iio_viral_status: Indicates the IIO cluster had gone to viral. This bit has no effect on hardware and does not indicate the IIO is currently in the viral state. This bit is persistent through warm reset (sticky), even though the viral state is not.
2:2	RW	0x0	iio_global_viral_mask: 0: IIO Viral State assertion will cause IIO hardware packet blocking. 1: IIO Viral State assertion will not cause IIO hardware packet blocking.
1:1	RW	0x0	Reserved (Rsvd): Reserved
0:0	RW	0x0	iio_fatal_viral_alert_enable: Enables IIO viral alert.

7.7.32 errpinctl

This register provides the option to configure an error pin to either as a special purpose error pin which is asserted based on the detected error severity, or as a general purpose output which is asserted based on the value in the ERRPINDAT. The assertion of the error pins can also be completely disabled by this register.

Type: CFG **Bus:** 0 **Offset:** 0xa4		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
5:4	RW	0x0	pin2: 11: Reserved. 10: Assert Error Pin when error severity 2 is set in the system event status reg. 01: Assert and De-assert Error pin according to error pin data register. 00: Disable Error pin assertion
3:2	RW	0x0	pin1: 11: Reserved. 10: Assert Error Pin when error severity 1 is set in the system event status reg. 01: Assert and De-assert Error pin according to error pin data register. 00: Disable Error pin assertion
1:0	RW	0x0	pin0: 11: Reserved. 10: Assert Error Pin when error severity 0 is set in the system event status reg. 01: Assert and De-assert Error pin according to error pin data register. 00: Disable Error pin assertion

7.7.33 errpinsts

This register reflects the state of the error pin assertion. The status bit of the corresponding error pin is set upon the deassertion to assertion transition of the error pin. This bit is cleared by the software with writing 1 to the corresponding bit.

Type: CFG	**PortID:** N/A		
Bus: 0	**Device:** 5	**Function:** 2	
Offset: 0xa8			

Bit	Attr	Default	Description
2:2	RW1CS	0x0	pin2: This bit is set upon the transition of deassertion to assertion of the Error pin. Software write 1 to clear the status.
1:1	RW1CS	0x0	pin1: This bit is set upon the transition of deassertion to assertion of the Error pin. Software write 1 to clear the status.
0:0	RW1CS	0x0	pin0: This bit is set upon the transition of deassertion to assertion of the Error pin. Software write 1 to clear the status.

7.7.34 errpindat

This register provides the data value when the error pin is configured as a general purpose output.

Type: CFG	**PortID:** N/A		
Bus: 0	**Device:** 5	**Function:** 2	
Offset: 0xac			

Bit	Attr	Default	Description
2:2	RW-LB	0x0	pin2: This bit acts as the general purpose output for the Error[2] pin. Software setsclears this bit to assertdeassert Error[2] pin. This bit applies only when ERRPINCTL[5:4] = 01; otherwise it is reserved. 0: Assert ERR#2 pin drive low 1: De-assert ERR#2 pin float high **Notes:** • This pin is open drain and must be pulled high by external resistor when deasserted. • The BIOS needs to write 1 to this bit for security reasons if this register is not used.
1:1	RW-LB	0x0	pin1: This bit acts as the general purpose output for the Error[1] pin. Software setsclears this bit to assertdeassert Error[1] pin. This bit applies only when ERRPINCTL[3:2] = 01; otherwise it is reserved. 0: Assert ERR#1 pin drive low 1: De-assert ERR#1 pin float high **Notes:** • This pin is open drain and must be pulled high by external resistor when deasserted. • The BIOS needs to write 1 to this bit for security reasons if this register is not used.

Type: CFG **Bus:** 0 **Offset:** 0xac	**PortID:** N/A **Device:** 5		**Function:** 2
Bit	**Attr**	**Default**	**Description**
0:0	RW-LB	0x0	pin0: This bit acts as the general purpose output for the Error[0] pin. Software setsclears this bit to assertdeassert Error[0] pin. This bit applies only when ERRPINCTL[1:0] = 01; otherwise it is reserved. 0: Assert ERR#0 pin drive low 1: De-assert ERR#0 pin float high **Notes:** • This pin is open drain and must be pulled high by external resistor when deasserted. • The BIOS needs to write 1 to this bit for security reasons if this register is not used.

7.7.35 vppctl

This register defines the control/command for PCA9555.

Type: CFG **Bus:** 0 **Offset:** 0xb0	**PortID:** N/A **Device:** 5		**Function:** 2
Bit	**Attr**	**Default**	**Description**
63:60	RO	0x1	vpp_version: Specified the version of this structure for BIOS use. 0: VPPCTL with 11 PCIe ports. 1: VPPCTL with 11 PCIe prots + VPPMEM with 4 memory ports.
59:56	RV	0x0	Reserved
55:55	RWS	0x0	vpp_reset_mode: 0: Power good reset will reset the VPP state machines and hard reset will cause the VPP state machine to terminate at the next 'logical' VPP stream boundary and then reset the VPP state machines 1: Both power good and hard reset will reset the VPP state machines
54:44	RWS	0x0	vpp_en: When set, the VPP function for the corresponding root port is enabled. Enable Root Port [54] Port 3d [53] Port 3c [52] Port 3b [51] Port 3a [50] Port 2d [49] Port 2c [48] Port 2b [47] Port 2a [46] Port 1b [45] Port 1a [44] Port 0 (PCIe mode only)

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xb0**				

Bit	Attr	Default	Description
43:0	RWS	0x0	vpp_enaddr: Assigns the VPP address of the device on the VPP interface and assigns the port address for the ports within the VPP device. There are more address bits then root ports so assignment must be spread across VPP ports. Port Addr Root Port [40] [43:41] Port 3d [36] [39:37] Port 3c [32] [35:33] Port 3b [28] [31:29] Port 3a [24] [27:25] Port 2d [20] [23:21] Port 2c [16] [19:17] Port 2b [12] [15:13] Port 2a [8] [11:9] Port 1a [4] [7:5] Port 1a [0] [3:1] Port 0 (PCIe mode only)

7.7.36 vppsts

This register defines the status from PCA9555

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xb8**				

Bit	Attr	Default	Description
0:0	RW1CS	0x0	vpp_error: VPP Port error happened that is, an unexpected STOP of NACK was seen on the VPP port

7.7.37 vppfreq

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xbc**				

Bit	Attr	Default	Description
31:24	RWS	0x1e	vpp_tpf: Pulse Filter should be set to 60 nseconds. The value used is dependent on the internal clock frequency. In this case, internal clock frequency is 500 MHz, so the default value represents 60 nseconds at that rate.
23:16	RWS	0x96	vpp_thd_data: Hold time for Data is 300nS. The default value is set to 300nS when the internal clock rate is 500MHz.
11:0	RWS	0x9c4	vpp_tsu_thd: Represents the high time and low time of the SCL pin. It should be set to 5uS for a 100 KHz SCL clock 5 µseconds high time and 5 µseconds low time. The default value represents 5 µseconds with an internal clock of 500 MHz.

7.7.38 vppmem

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xc0**				

Bit	Attr	Default	Description
63:40	RV	0x0	Reserved:
39:32	RWS	0x0	vpp_en: When set, the VPP function for the corresponding root port is enabled. Enable Root Port [39] reserved. [38] reserved. [37] reserved. [36] reserved. [35] Memory Channel x [34] Memory Channel x [33] Memory Channel x [32] Memory Channel x
31:0	RWS	0x0	vpp_enaddr: Assigns the VPP address of the device on the VPP interface and assigns the port address for the ports within the VPP device. There are for memory channel hot-plug. Port Addr Root Port [31] [30:28] Reserved [27] [27:24] Reserved [23] [22:20] Reserved [19] [18:16] Reserved [15] [14:12] Memory Channel x [11] [10:8] Memory Channel x [7] [6:4] Memory Channel x [3] [2:0] Memory Channel x

7.7.39 vpp_inverts

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0xc8**				

Bit	Attr	Default	Description
2:2	RWS	0x0	dfr_inv_mrl: Inverts the MRL signal
1:1	RWS	0x0	dfr_inv_emil: Inverts the EMIL signal
0:0	RWS	0x0	dfr_inv_pwren: Inverts the PWREN signal

7.7.40 miscprivc

Type: CFG Bus: 0 Offset: 0x16c		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:0	RWS	0x0	notused:

7.7.41 gcerrst

This register indicates the corrected error reported to the IIO global error logic. An individual error status bit that is set indicates that a particular local interface has detected an error.

Type: CFG Bus: 0 Offset: 0x1a8		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:27	RV	0x0	RSVD
26:26	RV	0x0	**iMC error** Memory Controller Error Status. **Note**: This bit is only available for the processor B0 or later steppings. For A steppings, the bit is Reserved.mi:
25:25	RW	0b	Intel® VT-d Error
24:24	RW	0b	Miscellaneous Error
23:23	RW	0b	IIO Core Error
22:22	RW	0b	Reserved
21:21	RW	0b	Reserved
20:20	RW	0b	DMI Error
19:16	RV	0x0	Reserved
15:5	RW	0x0	PCIe* Error Bit 5: Port 0 Bit 6: Port 1a Bit 7: Port 1b Bit 8: Port 2a Bit 9: Port 2b Bit 10: Port 2c Bit 11: Port 2d Bit 12: Port 3a Bit 13: Port 3b Bit 14: Port 3c Bit 15: Port 3d
4:2	RV	0x0	Reserved
1:1	RW	0x0	IRP1 Error Mask
0:0	RW	0b	IRP0 Error Mask; When set, disables logging of error

7.7.42 gcferrst

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1a8**				

Bit	Attr	Default	Description
31:27	RV	0x0	RSVD
26:26	RV	0x0	**iMC error** Memory Controller Error Status. **Note**: This bit is only available for the processor B0 or later steppings. For A steppings, the bit is Reserved.mi:
25:25	RW	0b	Intel® VT-d Error
24:24	RW	0b	Miscellaneous Error
23:23	RW	0b	IIO Core Error
22:22	RW	0b	Reserved
21:21	RW	0b	Reserved
20:20	RW	0b	DMI Error
19:16	RV	0x0	Reserved
15:5	RW	0x0	PCIe* Error Bit 5: Port 0 Bit 6: Port 1a Bit 7: Port 1b Bit 8: Port 2a Bit 9: Port 2b Bit 10: Port 2c Bit 11: Port 2d Bit 12: Port 3a Bit 13: Port 3b Bit 14: Port 3c Bit 15: Port 3d
4:2	RV	0x0	Reserved
1:1	RW	0x0	IRP1 Error Mask
0:0	RW	0b	IRP0 Error Mask; When set, disables logging of error

7.7.43 gcnerrst

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1a8**				

Bit	Attr	Default	Description
31:27	RV	0x0	RSVD
26:26	RV	0x0	**iMC error** Memory Controller Error Status. **Note**: This bit is only available for the processor B0 or later steppings. For A steppings, the bit is Reserved.mi:
25:25	RW	0b	Intel® VT-d Error
24:24	RW	0b	Miscellaneous Error
23:23	RW	0b	IIO Core Error
22:22	RW	0b	Reserved
21:21	RW	0b	Reserved

Type: CFG Bus: 0 Offset: 0x1a8		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
20:20	RW	0b	DMI Error
19:16	RV	0x0	Reserved
15:5	RW	0x0	PCIe* Error Bit 5: Port 0 Bit 6: Port 1a Bit 7: Port 1b Bit 8: Port 2a Bit 9: Port 2b Bit 10: Port 2c Bit 11: Port 2d Bit 12: Port 3a Bit 13: Port 3b Bit 14: Port 3c Bit 15: Port 3d
4:2	RV	0x0	Reserved
1:1	RW	0x0	IRP1 Error Mask
0:0	RW	0b	IRP0 Error Mask; When set, disables logging of error

7.7.44 gnerrst

Global Non-Fatal Error Status.

This register indicates the non-fatal error reported to the IIO global error logic. An individual error status bit that is set indicates that a particular local interface has detected an error.

Type: CFG Bus: 0 Offset: 0x1c0		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
25:25	RW1CS	0x0	vtd:
24:24	RW1CS	0x0	mi:
23:23	RW1CS	0x0	iio: 1
22:22	RW1CS	0x0	dma: This bit indicates that IIO has detected an error in its DMA engine.
21:21	RW1CS	0x0	thermal: 1
20:20	RW1CS	0x0	dmi: This bit indicates that IIO DMI port 0 has detected an error.
15:15	RW1CS	0x0	pcie10: 1
14:14	RW1CS	0x0	pcie9: 1
13:13	RW1CS	0x0	pcie8: 1

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1c0**				

Bit	Attr	Default	Description
12:12	RW1CS	0x0	pcie7: 1
11:11	RW1CS	0x0	pcie6: 1
10:10	RW1CS	0x0	pcie5: 1
9:9	RW1CS	0x0	pcie4: 1
8:8	RW1CS	0x0	pcie3: 1
7:7	RW1CS	0x0	pcie2: 1
6:6	RW1CS	0x0	pcie1: 1
5:5	RW1CS	0x0	pcie0: 1
3:3	RW1CS	0x0	csipro1: 1
2:2	RW1CS	0x0	csipro0: 1
1:1	RW1CS	0x0	csi1_err: 1
0:0	RW1CS	0x0	csi0_err: 1

7.7.45 gferrst

Global Fatal Error Status.

This register indicates the fatal error reported to the IIO global error logic. An individual error status bit that is set indicates that a particular local interface has detected an error.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1c4**				

Bit	Attr	Default	Description
25:25	RW1CS	0x0	vtd: This register indicates the fatal error reported to the Intel VT-d error logic. An individual error status bit that is set indicates that a particular local interface has detected an error.
24:24	RW1CS	0x0	mi: 1
23:23	RW1CS	0x0	iio: 1
22:22	RW1CS	0x0	dma: This bit indicates that IIO has detected an error in its DMA engine.
21:21	RW1CS	0x0	thermal: 1
20:20	RW1CS	0x0	dmi: This bit indicates that IIO DMI port 0 has detected an error.
15:15	RW1CS	0x0	pcie10: 1
14:14	RW1CS	0x0	pcie9: 1
13:13	RW1CS	0x0	pcie8: 1
12:12	RW1CS	0x0	pcie7: 1
11:11	RW1CS	0x0	pcie6: 1
10:10	RW1CS	0x0	pcie5: 1
9:9	RW1CS	0x0	pcie4: 1
8:8	RW1CS	0x0	pcie3: 1
7:7	RW1CS	0x0	pcie2: 1
6:6	RW1CS	0x0	pcie1: 1
5:5	RW1CS	0x0	pcie0: 1
3:3	RW1CS	0x0	csipro1: 1

Type: CFG **Bus:** 0 **Offset:** 0x1c4		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
2:2	RW1CS	0x0	csipro0: 1
1:1	RW1CS	0x0	tras_csi1: 1
0:0	RW1CS	0x0	tras_csi0: 1

7.7.46 gerrctl

Global Error Control.

This register controls/masks the reporting of errors detected by the IIO local interfaces. An individual error control bit that is set masks error reporting of the particular local interface; software may set or clear the control bit. This register is sticky and can only be reset by PWRGOOD.

Note: Bit fields in this register can become reserved depending on the port configuration. For example, if the PCI-E port is configured as 2X8 ports, then only the corresponding PCI-EX8 bit fields are valid; other bits are unused and reserved.Global error control register masks errors reported from the local interface to the global register. If the an error reporting is disabled in this register, all errors from the corresponding local interface will not set any of the global error status bits.

Type: CFG **Bus:** 0 **Offset:** 0x1c8		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
26:26	RV	0x0	**iMC error** Memory Controller Error Status. **Note**: This bit is only available for the processor B0 or later stepping. For A steppings, the bit is Reserved.
25:25	RW	0x0	vtd_err_msk:
24:24	RW	0x0	mi_err_msk:
23:23	RW	0x0	iio_err_msk: 1
21:21	RW	0x0	therm_err_msk: 1
20:20	RW	0x0	dmi_err_msk: This bit enables/masks the error detected in the DMI[0] Port.
15:15	RW	0x0	pcie_err_msk10: 1
14:14	RW	0x0	pcie_err_msk9: 1
13:13	RW	0x0	pcie_err_msk8: 1
12:12	RW	0x0	pcie_err_msk7: 1

Type: CFG Bus: 0 Offset: 0x1c8	PortID: N/A Device: 5	Function: 2	
Bit	**Attr**	**Default**	**Description**
11:11	RW	0x0	pcie_err_msk6: 1
10:10	RW	0x0	pcie_err_msk5: 1
9:9	RW	0x0	pcie_err_msk4: 1
8:8	RW	0x0	pcie_err_msk3: 1
7:7	RW	0x0	pcie_err_msk2: 1
6:6	RW	0x0	pcie_err_msk1: 1
5:5	RW	0x0	pcie_err_msk0: 1
3:3	RW	0x0	csip_err_msk1: 1
2:2	RW	0x0	csip_err_msk0: 1
1:1	RW	0x0	csi_err_msk1:
0:0	RW	0x0	csi_err_msk0: When set, disables logging of this error

7.7.47 gsysst

Global System Event Status.

This register indicates the error severity signaled by the IIO global error logic. Setting of an individual error status bit indicates that the corresponding error severity has been detected by the IIO.

Type: CFG Bus: 0 Offset: 0x1cc	PortID: N/A Device: 5	Function: 2	
Bit	**Attr**	**Default**	**Description**
4:4	ROS_V	0x0	sev4: Thermal Trip Error
3:3	ROS_V	0x0	sev3: Thermal Alert Error
2:2	ROS_V	0x0	sev2: When set, IIO has detected an error of error severity 2
1:1	ROS_V	0x0	sev1: When set, IIO has detected an error of error severity 1
0:0	ROS_V	0x0	sev0: When set, IIO has detected an error of error severity 0

7.7.48 gsysctl

Global System Event Control.

The system event control register controls/masks the reporting the errors indicated by the system event status register. When cleared, the error severity does not cause the generation of the system event. When set, detection of the error severity generates system events according to system event map register (SYSMAP).

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1d0**				

Bit	Attr	Default	Description
4:4	RW	0x0	**7.7.49 sev4_en:** Thermal Trip Enable
3:3	RW	0x0	sev3_en: Thermal Alert Enable
2:2	RW	0x0	sev2_en:
1:1	RW	0x0	sev1_en:
0:0	RW	0x0	sev0_en:

7.7.50 gtime_lsb

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1d4**				

Bit	Attr	Default	Description
31:0	RWS_V	0x0	gtime_lsb:

7.7.51 gtime_msb

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x1d8**				

Bit	Attr	Default	Description
31:0	RWS_V	0x0	gtime_msb:

7.7.52 gfferrst, gfnerrst

Global Fatal FERR and NERR Status.

Type: CFG **Bus:** 0 **Offset:** 0x1dc, 0x1e8	**PortID:** N/A **Device:** 5	**Function:** 2	
Bit	**Attr**	**Default**	**Description**
26:0	ROS_V	0x0	log: This field logs the global error status register content when the first fatal error is reported. This has the same format as the global fatal error status register (GFERRST).

7.7.53 gnferrst, gnnerrst

Global Non-Fatal FERR and NERR Status

Type: CFG **Bus:** 0 **Offset:** 0x1ec, 0x1f8	**PortID:** N/A **Device:** 5	**Function:** 2	
Bit	**Attr**	**Default**	**Description**
26:0	ROS_V	0x0	log: This filed logs the global error status register content when the first non-fatal error is reported. This has the same format as the global non-fatal error status register (GNERRST).

7.7.54 irpp[0:1]errst

IRP Protocol Error Status.

This register indicates the error detected by the Coherent Interface.

Type: CFG **Bus:** 0 **Offset:** 0x230, 0x2b0	**PortID:** N/A **Device:** 5	**Function:** 2	
Bit	**Attr**	**Default**	**Description**
14:14	RW1CS	0x0	protocol_parity_error: (DB) Originally used for detecting parity error on coherent interface, however, no parity checks exist. So this logs parity errors on data from the IIO switch on the inbound path.
13:13	RW1CS	0x0	protocol_qt_overflow_underflow: (DA)
10:10	RW1CS	0x0	protocol_rcvd_unexprsp: (D7) A completion has been received from the Coherent Interface that was unexpected.
4:4	RW1CS	0x0	csr_acc_32b_unaligned: (C3)

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x230, 0x2b0**				

Bit	Attr	Default	Description
3:3	RW1CS	0x0	wrcache_uncecc_error: (C2) A double bit ECC error was detected within the Write Cache.
2:2	RW1CS	0x0	protocol_rcvd_poison: (C1) A poisoned packet has been received from the Coherent Interface.
1:1	RW1CS	0x0	wrcache_correcc_error: (B4) A single bit ECC error was detected and corrected within the Write Cache.

7.7.55 irpp[0:1]errctl

IRP Protocol Error Control.

This register enables the error status bit setting for a Coherent Interface detected error. Setting of the bit enables the setting of the corresponding error status bit in IRPPERRST register. If the bit is cleared, the corresponding error status will not be set.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x234, 0x2b4**				

Bit	Attr	Default	Description
14:14	RWS	0x0	protocol_parity_error: (DB) 0: Disable error status logging for this error 1: Enable Error status logging for this error
13:13	RWS	0x0	protocol_qt_overflow_underflow: (DA) 0: Disable error status logging for this error 1: Enable Error status logging for this error
10:10	RWS	0x0	protocol_rcvd_unexprsp: (D7) 0: Disable error status logging for this error 1: Enable Error status logging for this error
4:4	RWS	0x0	csr_acc_32b_unaligned: (C3) 0: Disable error status logging for this error 1: Enable Error status logging for this error
3:3	RWS	0x0	wrcache_uncecc_error: (C2) 0: Disable error status logging for this error 1: Enable Error status logging for this error
2:2	RWS	0x0	protocol_rcvd_poison: (C1) 0: Disable error status logging for this error 1: Enable Error status logging for this error
1:1	RWS	0x0	wrcache_correcc_error: (B4) 0: Disable error status logging for this error 1: Enable Error status logging for this error

7.7.56 irpp[0:1]fferrst, irpp[0:1]fnerrst

IRP Protocol Fatal FERR and NERR Status.

The error status log indicates which error is causing the report of the first fatal error event.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**irp0: 0x238, 0x23c** **irp1: 0x2b8, 0x2bc**				

Bit	Attr	Default	Description
14:14	ROS_V	0x0	protocol_parity_error: (DB) Originally used for detecting parity error on coherent interface, however, no parity checks exist. So this logs parity errors on data from the IIO switch on the inbound path.
13:13	ROS_V	0x0	protocol_qt_overflow_underflow: (DC)
10:10	ROS_V	0x0	protocol_rcvd_unexprsp: (D7) A completion has been received from the Coherent Interface that was unexpected.
4:4	ROS_V	0x0	csr_acc_32b_unaligned: (C3)
3:3	ROS_V	0x0	wrcache_uncecc_error: (C2) A double bit ECC error was detected within the Write Cache.
2:2	ROS_V	0x0	protocol_rcvd_poison: (C1) A poisoned packet has been received from the Coherent Interface.
1:1	ROS_V	0x0	wrcache_correcc_error: (B4) A single bit ECC error was detected and corrected within the Write Cache.

7.7.57 irpp[0:1]fferrhd[0:3]

IRP Protocol Fatal FERR Header Log.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**irpp0fferrhd: 0x240, 0x244, 0x248, 0x24c** **irpp1fferrhd: 0x2c0, 0x2c4, 0x2c8, 0x2cc**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	hdr: Logs the first Dword of the header on an error condition

7.7.58 irpp[0:1]nferrst, irpp[0:1]nnerrst

IRP Protocol Non-Fatal FERR and NERR Status.

The error status log indicates which error is causing the report of the first non-fatal error event.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**irp0: 0x250, 0x254,** **irp1: 0x2d0, 0x2d4**				

Bit	Attr	Default	Description
14:14	ROS_V	0x0	protocol_parity_error: Originally used for detecting parity error on coherent interface, however, no parity checks exist. So this logs parity errors on data from the IIO switch on the inbound path.
13:13	ROS_V	0x0	protocol_qt_overflow_underflow:
10:10	ROS_V	0x0	protocol_rcvd_unexprsp: A completion has been received from the Coherent Interface that was unexpected.
4:4	ROS_V	0x0	csr_acc_32b_unaligned:
3:3	ROS_V	0x0	wrcache_uncecc_error: A double bit ECC error was detected within the Write Cache.
2:2	ROS_V	0x0	protocol_rcvd_poison: A poisoned packet has been received from the Coherent Interface.
1:1	ROS_V	0x0	wrcache_correcc_error: A single bit ECC error was detected and corrected within the Write Cache.

7.7.59 irpp[0:1]nferrhd[0:3]

IRP Protocol Non-Fatal FERR Header Log.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**irpp0nferrhd: 0x258, 0x25c, 0x260, 0x264** **irpp1nferrhd: 0x2d8, 0x2dc, 0x2e0, 0x2e4**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	hdr: Logs the first Dword of the header on an error condition

7.7.60 irpp[0:1]errcntsel

IRP Protocol Error Counter Select.

Type: CFG **Bus:** 0 **Offset:** 0x268, 0x2e8	**PortID:** N/A **Device:** 5		**Function:** 2
Bit	**Attr**	**Default**	**Description**
18:0	RW	0x0	irp_error_count_select: See IRPP0ERRST for per bit description of each error. Each bit in this field has the following behavior: 0: Do not select this error type for error counting 1: Select this error type for error counting

7.7.61 irpp[0:1]errcnt

IRP Protocol Error Count.

Type: CFG **Bus:** 0 **Offset:** 0x26c, 0x2ec	**PortID:** N/A **Device:** 5		**Function:** 2
Bit	**Attr**	**Default**	**Description**
7:7	RW1CS	0x0	errovf: Error Accumulator Overflow 0: No overflow occurred 1: Error overflow. The error count may not be valid.
6:0	RW1CS	0x0	errcnt: This counter accumulates errors that occur when the associated error type is selected in the ERRCNTSEL register. **Notes:** • This register is cleared by writing 7Fh. • Maximum counter available is 127d 7Fh

7.7.62 iioerrst

IIO Core Error Status.

This register indicates the IIO internal core errors detected by the IIO error logic. An individual error status bit that is set indicates that a particular error occurred; software may clear an error status by writing a 1 to the respective bit. This register is sticky and can only be reset by PWRGOOD. Clearing of the IIOERRST is done by clearing the corresponding IIOERRST bits.

Type: CFG **Bus:** 0 **Offset:** 0x300		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
31:7	RO	0x0	reserved: 1
6:6	RW1CS	0x0	c6:
5:5	RW1CS	0x0	RSVD
4:4	RW1CS	0x0	c4:
3:3	RW1CS	0x0	unused3:
2:2	RW1CS	0x0	unused2:
1:1	RW1CS	0x0	unused1:
0:0	RW1CS	0x0	unused0:

7.7.63 iioerrctl

IIO Core Error Control.

This register controls the reporting of IIO internal core errors detected by the IIO error logic. An individual error control bit that is cleared masks reporting of that a particular error; software may set or clear the respective bit. This register is sticky and can only be reset by PWRGOOD.

Type: CFG **Bus:** 0 **Offset:** 0x304		**PortID:** N/A **Device:** 5	**Function:** 2
Bit	**Attr**	**Default**	**Description**
8:8	RWS_L	0x0	c4_inbound_ler_disable: Disable logging C4 error due to the PCIE being down due to being in LER mode. **Note:** Locked by RSPLCK
7:7	RWS_L	0x0	c4_outbound_ler_disable: Disable logging C4 error due to the PCIE being down due to being in LER mode. **Note:** Locked by RSPLCK
6:6	RWS_L	0x0	c6: **Note:** Locked by RSPLCK

Type: CFG Bus: 0 Offset: 0x304		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
5:5	RWS_L	0x0	RSVD
4:4	RWS_L	0x0	c4: **Note:** Locked by RSPLCK
3:3	RWS_L	0x0	unused3: **Note:** Locked by RSPLCK
2:2	RWS_L	0x0	unused2: **Note:** Locked by RSPLCK
1:1	RWS_L	0x0	unused1: **Note:** Locked by RSPLCK
0:0	RWS_L	0x0	unused0: **Note:** Locked by RSPLCK

7.7.64 iioofferrst, iiofnerrst

IIO Core Fatal FERR and NERR Status

Type: CFG Bus: 0 Offset: 0x308, 0x31c		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
6:0	ROS_V	0x0	iio_core_error_status_log: The error status log indicates which error is causing the report of the first error event. The encoding indicates the corresponding bit position of the error in the error status register.

7.7.65 iiofferrhd_[0:3]

IIO Core Fatal FERR Header.

Type: CFG Bus: 0 Offset: 0x30c, 0x310, 0x314, 0x318		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
31:0	ROS_V	0x0	iio_core_error_header_log: Logs the first Dword of the header on an error condition.

7.7.66 iionferrst, iionnerrst

IIO Core Non-Fatal FERR and NERR Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x320, 0x334**				

Bit	Attr	Default	Description
6:0	ROS_V	0x0	iio_core_error_status_log: The error status log indicates which error is causing the report of the first error event. The encoding indicates the corresponding bit position of the error in the error status register.

7.7.67 iionferrhd_[0:3]

IIO Core Non-Fatal FERR Header.

Header log stores the IIO data path header information of the associated IIO core error. The header indicates where the error is originating from and the address of the cycle.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x324, 0x328, 0x32c, 0x330**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	iio_core_error_header_log: The error status log indicates which error is causing the report of the first error event. The encoding indicates the corresponding bit position of the error in the error status register.

7.7.68 iioerrcntsel

IIO Core Error Counter Selection.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x33c**				

Bit	Attr	Default	Description
6:6	RW-L	0x0	c6:
5:5	RW-L	0x0	c5:
4:4	RW-L	0x0	c4:
3:3	RW-L	0x0	thirteen_msi_address_error_select: 1
2:2	RW-L	0x0	twofive_core_header_queue_parity_error_select: 1
1:1	RW-L	0x0	onetwo_dma_or_vt_d_access_xing_64_bit_boundary_error_select: 1
0:0	RW-L	0x0	reserved: 1

7.7.69 iioerrcnt

IIO Core Error Counter.

Type: CFG Bus: 0 Offset: 0x340		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
7:7	RW1CS	0x0	errovf: 0: No overflow occurred1: Error overflow. The error count may not be valid.
6:0	RW1CS	0x0	errcnt: This counter accumulates errors that occur when the associated error type is selected in the ERRCNTSEL register. **Notes:** • This register is cleared by writing 7Fh. • Maximum counter available is 127d (7Fh).

7.7.70 mierrst

Miscellaneous Error Status

Type: CFG Bus: 0 Offset: 0x380		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
4:4	RW1CS	0x0	RSVD
3:3	RW1CS	0x0	vpp_err_sts:
2:2	RW1CS	0x0	RSVD
1:1	RW1CS	0x0	RSVD
0:0	RW1CS	0x0	RSVD

7.7.71 mierrctl

Miscellaneous Error Control.

Type: CFG Bus: 0 Offset: 0x384		PortID: N/A Device: 5	Function: 2
Bit	**Attr**	**Default**	**Description**
4:4	RWS	0x0	dfx_inj_err:
3:3	RWS	0x0	vpp_err_sts:
2:2	RWS	0x0	jtag_tap_sts:
1:1	RWS	0x0	smbus_port_sts: This bit has no effect.
0:0	RWS	0x0	cfg_reg_par:

7.7.72 mifferrst, mifnerrst

Miscellaneous Fatal FERR and NERR Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x388, 0x39c**				

Bit	Attr	Default	Description
10:0	ROS_V	0x0	mi_err_st_log:

7.7.73 mifferrhdr_[0:3]

Miscellaneous Fatal FERR Header Log.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x38c, 0x390, 0x394, 0x398**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	hdr:

7.7.74 minferrst, minnerrst

Miscellaneous Non-Fatal FERR and NERR Status.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x3a0, 0x3b4**				

Bit	Attr	Default	Description
10:0	ROS_V	0x0	mi_err_st_log:

7.7.75 minferrhdr_[0:3]

Miscellaneous Non-Fatal FERR Header Log.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**2**
Offset:	**0x3a4, 0x3a8, 0x3ac, 0x3b0**				

Bit	Attr	Default	Description
31:0	ROS_V	0x0	hdr:

7.7.76 mierrcntsel

Miscellaneous Error Count Select.

Type: CFG **Bus:** 0 **Offset:** 0x3bc	**PortID:** N/A **Device:** 5	**Function:** 2	
Bit	**Attr**	**Default**	**Description**
4:4	RW	0x0	dfx_inj_err:
3:3	RW	0x0	vpp_err_sts:
2:2	RW	0x0	jtag_tap_sts:
1:1	RW	0x0	smbus_port_sts: This bit has no effect.
0:0	RW	0x0	cfg_reg_par:

7.7.77 mierrcnt

Miscellaneous Error Count.

Type: CFG **Bus:** 0 **Offset:** 0x3c0	**PortID:** N/A **Device:** 5	**Function:** 2	
Bit	**Attr**	**Default**	**Description**
7:7	RW1CS	0x0	errovflow: 0: No overflow occurred1: Error overflow. The error count may not be valid.
6:0	RW1CS	0x0	errcnt: This counter accumulates errors that occur when the associated error type is selected in the ERRCNTSEL register. **Notes:** • This register is cleared by writing 7Fh. • Maximum counter available is 127d (7Fh).

7.8 Device 5 Function 4

I/OxAPCI Configuration Space.

Table 7-10. Integrated I/O Device 5 Function 4 Register Address Map

Register Name	Offset	Size
vid	0x0	16
did	0x2	16
pcicmd	0x4	16
pcists	0x6	16
rid	0x8	8
ccr	0x9	24
clsr	0xc	8
hdr	0xe	8
mbar	0x10	32
svid	0x2c	16
sid	0x2e	16
capptr	0x34	8
intlin	0x3c	8
intpin	0x3d	8
abar	0x40	16
pxpcap	0x44	32
snapshot_index	0x80	8
snapshot_window	0x90	32
ioapictetpc	0xa0	32
pmcap	0xe0	32
pmcsr	0xe4	32
ioadsels0	0x288	32
ioadsels1	0x28c	32
iointsrc0	0x2a0	32
iointsrc1	0x2a4	32
ioremintcnt	0x2a8	32
ioremgpecnt	0x2ac	32
FauxGV	0x2c4	32

7.8.1 vid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x0**				

Bit	Attr	Default	Description
15:0	RO	0x8086	vendor_identification_number: 1

7.8.2 did

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2**				

Bit	Attr	Default	Description
15:0	RO	0xe2c	device_identification_number: 1

7.8.3 pcicmd

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x4**				

Bit	Attr	Default	Description
10:10	RO	0x0	intxdisable: 1
9:9	RO	0x0	fb2be: 1
8:8	RO	0x0	serre: 1
7:7	RO	0x0	idsel: 1
6:6	RO	0x0	perrrsp: 1
5:5	RO	0x0	vga: 1
4:4	RO	0x0	memwrinv: 1
3:3	RO	0x0	spcen: 1
2:2	RW	0x0	bme: 1
1:1	RW	0x0	mse: 1
0:0	RO	0x0	iose: 1

7.8.4 pcists

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x6**				

Bit	Attr	Default	Description
15:15	RO-V	0x0	dpe: 1
14:14	RO	0x0	sse: 1
13:13	RO	0x0	rma: 1
12:12	RO	0x0	rta: 1
11:11	RW1C	0x0	sta: 1
10:9	RO	0x0	devselt: 1
8:8	RO	0x0	medierr: 1
7:7	RO	0x0	fb2bcap: 1
5:5	RO	0x0	sixtysixmhzcap: 1
4:4	RO	0x1	capl: 1
3:3	RO	0x0	intxst: 1

7.8.5 rid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x8**				

Bit	Attr	Default	Description
7:0	RO-V	0x0	revision_id: Reflects the Uncore Revision ID after reset. Reflects the Compatibility Revision ID after the BIOS writes 0x69 to any RID register in any processor function. **Implementation Note:** Read and write requests from the host to any RID register in any processor function are re-directed to the IIO cluster. Accesses to the CCR field are also redirected due to Dword alignment. It is possible that JTAG accesses are direct, so will not always be redirected.

7.8.6 ccr

Type: CFG Bus: 0 Offset: 0x9		PortID: N/A Device: 5	Function: 4
Bit	**Attr**	**Default**	**Description**
23:16	RO-V	0x80	base_class: Generic Device
15:8	RO-V	0x0	sub_class: Generic Device
7:0	RO-V	0x20	interface: 1

7.8.7 clsr

Type: CFG Bus: 0 Offset: 0xc		PortID: N/A Device: 5	Function: 4
Bit	**Attr**	**Default**	**Description**
7:0	RW	0x0	clsr_reg: 1

7.8.8 hdr

Type: CFG Bus: 0 Offset: 0xe		PortID: N/A Device: 5	Function: 4
Bit	**Attr**	**Default**	**Description**
7:7	RO	0x1	multi_function_device: This bit defaults to 1b since all these devices are multi-function
6:0	RO	0x0	configuration_layout: This field identifies the format of the configuration header layout. It is Type 0 for all these devices. The default is 00h, indicating a 'endpoint device'.

7.8.9 mbar

I/OxAPIC Based Address

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x10**				

Bit	Attr	Default	Description
31:12	RW	0x0	bar: This marks the 4KB aligned 32-bit base address for memory-mapped registers of I/OxAPICSide **Note**: Any accesses by means of message channel or JTAG minimum port to registers pointed to by the MBAR address, are not gated by MSE bit (in PCICMD register) being set that is, even if MSE bit is a 0, message channel accesses to the registers pointed to by MBAR address are allowed completed normally. These accesses are accesses from internal µcode/pcode and JTAG and they are allowed to access the registers normally even if this bit is clear.
3:3	RO	0x0	prefetchable: The I/OxAPIC registers are not prefetchable.
2:1	RO	0x0	type: The IOAPIC registers can only be placed below 4G system address space.
0:0	RO	0x0	memory_space: This Base Address Register indicates memory space.

7.8.10 svid

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2c**				

Bit	Attr	Default	Description
15:0	RW-O	0x8086	svid_reg: The default value specifies Intel but can be set to any value once after reset.

7.8.11 sid

This value is used to identify a particular subsystem.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2e**				

Bit	Attr	Default	Description
15:0	RW-O	0x0	sid_reg: Assigned by the subsystem vendor to uniquely identify the subsytem.

7.8.12 capptr

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x34**				

Bit	Attr	Default	Description
7:0	RO	0x44	capability_pointer: Points to the first capability structure for the device which is the PCIe capability.

7.8.13 intlin

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x3c**				

Bit	Attr	Default	Description
7:0	RO	0x0	intlin_reg:

7.8.14 intpin

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x3d**				

Bit	Attr	Default	Description
7:0	RO	0x0	intpin_reg:

7.8.15 abar

I/OxAPIC Alternate BAR.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x40**				

Bit	Attr	Default	Description
15:15	RW	0x0	abar_enable: When set, the range FECX_YZ00 to FECX_YZFF is enabled as an alternate access method to the I/OxAPIC registers and these addresses are claimed by the IIO's internal I/OxAPIC regardless of the setting the MSE bit in the IOxAPIC config space. Bits 'XYZ' are defined below.Side **Note**: Any accesses by means of message channel or JTAG minimum port to registers pointed to by the ABAR address, are not gated by this bit being set that is, even if this bit is a 0, message channel accesses to the registers pointed to by ABAR address are allowed completed normally. These accesses are accesses from internal ucode/pcode and JTAG and they are allowed to access the registers normally even if this bit is clear.
11:8	RW	0x0	base_address_19: 16 (XBAD) These bits determine the high order bits of the I/O APIC address map. When a memory address is recognized by the IIO which matches FECX_YZ00-to-FECX_YZFF, the IIO will respond to the cycle and access the internal I/O APIC.
7:4	RW	0x0	base_address_15: 12 (YBAD) These bits determine the low order bits of the I/O APIC address map. When a memory address is recognized by the IIO which matches FECX_YZ00-to-FECX_YZFF, the IIO will respond to the cycle and access the internal I/O APIC.
3:0	RW	0x0	base_address_11: 8 (ZBAD) These bits determine the low order bits of the I/O APIC address map. When a memory address is recognized by the IIO which matches FECX_YZ00-to-FECX_YZFF, the IIO will respond to the cycle and access the internal I/O APIC.

7.8.16 pxpcap

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x44**				

Bit	Attr	Default	Description
29:25	RO	0x0	interrupt_message_numnber: 1
24:24	RO	0x0	slot_implemented:
23:20	RO	0x9	device_port_type: Device type is Root Complex Integrated Endpoint

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x44**				

Bit	Attr	Default	Description
19:16	RO	0x1	capability_version: PCI Express* Capability is Compliant with Version 1.0 of the PCI Express* Specification. **Note:** This capability structure is not compliant with Versions beyond 1.0, since they require additional capability registers to be reserved. The only purpose for this capability structure is to make enhanced configuration space available. Minimizing the size of this structure is accomplished by reporting version 1.0 compliancy and reporting that this is an integrated root port device. As such, only three Dwords of configuration space are required for this structure.
15:8	RO	0xe0	next_ptr: Pointer to the next capability. Set to 0 to indicate there are no more capability structures, else default value
7:0	RO	0x10	capability_idat: Provides the PCI Express* capability ID assigned by PCI-SIG.

7.8.17 snapshot_index

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x80**				

Bit	Attr	Default	Description
7:0	RW	0x0	ssidx:

7.8.18 snapshot_window

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x90**				

Bit	Attr	Default	Description
31:0	RO-V	0x0	sswindow:

7.8.19 ioapictetpc

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0xa0**				

Bit	Attr	Default	Description
16:16	RW	0x0	cbdma0_inta: 0: srcint is connected to IOAPIC table entry 7 1: srcint is connected to IOAPIC table entry 23 **Note:** This bit should never be set due to a bug.
12:12	RW	0x0	ntb_int: 0: srcint is connected to IOAPIC table entry 16 1: srcint is connected to IOAPIC table entry 23 **Note:** This bit was not used by RTL.
10:10	RW	0x0	port3c_intb: 0: srcint is connected to IOAPIC table entry 21 1: srcint is connected to IOAPIC table entry 19
8:8	RW	0x0	port3a_intb: 0: srcint is connected to IOAPIC table entry 20 1: srcint is connected to IOAPIC table entry 17
6:6	RW	0x0	port2c_intb: 0: srcint is connected to IOAPIC table entry 13 1: srcint is connected to IOAPIC table entry 11
4:4	RW	0x0	port2a_intb: 0: srcint is connected to IOAPIC table entry 12 1: srcint is connected to IOAPIC table entry 9
0:0	RW	0x0	port0_intb: 0: srcint is connected to IOAPIC table entry 1 1: srcint is connected to IOAPIC table entry 3

7.8.20 pmcap

Power Management Capabilities.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0xe0**				

Bit	Attr	Default	Description
31:27	RO	0x0	pme_support: Bits 31, 30 and 27 must be set to '1' for PCI-PCI bridge structures representing ports on root complexes.
26:26	RO	0x0	d2_support: I/OxAPIC does not support power management state D2
25:25	RO	0x0	d1_support: I/OxAPIC does not support power management state D1
24:22	RO	0x0	aux_current:
21:21	RO	0x0	device_specific_initalization:
19:19	RO	0x0	pme_clock: This field is hardwired to 0h as it does not apply to PCI Express*.

Type: CFG Bus: 0 Offset: 0xe0		PortID: N/A Device: 5	Function: 4
Bit	**Attr**	**Default**	**Description**
18:16	RW-O	0x3	version: This field is set to 3h (PM 1.2 compliant) as version number. Bit is RW-O to make the version 2h incase legacy OS'es have any issues.
15:8	RO	0x0	next_pointer: This is the last capability in the chain and hence set to 0.
7:0	RO	0x1	capability_id: Provides the PM capability ID assigned by PCI-SIG.

7.8.21 pmcsr

Power Management Control and Status.

Type: CFG Bus: 0 Offset: 0xe4		PortID: N/A Device: 5	Function: 4
Bit	**Attr**	**Default**	**Description**
31:24	RO	0x0	data: Not relevant for I/OxAPIC
23:23	RO	0x0	bpcce: Not relevant for I/OxAPIC
22:22	RO	0x0	b2b3: Not relevant for I/OxAPIC
15:15	RO	0x0	pmests: Not relevant for I/OxAPIC
14:13	RO	0x0	dscl: Not relevant for I/OxAPIC
12:9	RO	0x0	dsel: Not relevant for I/OxAPIC
8:8	RO	0x0	pmeen: Not relevant for I/OxAPIC
3:3	RO	0x1	rstd3hotd0: Indicates I/OxAPIC does not reset its registers when transitioning from D3hot to D0.
1:0	RW-V	0x0	power_state: This 2-bit field is used to determine the current power state of the function and to set a new power state as well. 00: D0 01: D1 (not supported by IOAPIC) 10: D2 (not supported by IOAPIC) 11: D3_hot If Software tries to write 01 or 10 to this field, the power state does not change from the existing power state (which is either D0 or D3hot) and nor do these bits1:0 change value. When in D3hot state, I/OxAPIC will a) respond to only Type 0 configuration transactions targeted at the device's configuration space, when in D3hot state c) will not respond to memory (that is, D3hot state is equivalent to MSE), accesses to MBAR region **Note**: ABAR region access still go through in D3hot state, if it enabled d) will not generate any MSI writes

7.8.22 ioadsels0

I/OxAPIC DSELS Register 0.

Type: CFG **Bus: 0** **Offset: 0x288**	**PortID: N/A** **Device: 5**	**Function: 4**

Bit	Attr	Default	Description
28:28	RWS	0x0	sw2ipc_aer_negedge_msk:
27:27	RWS	0x0	sw2ipc_aer_event_sel:
26:0	RWS	0x0	gttcfg2SIpcIOADels0: gttcfg2SIpcIOADels0[26:0]

7.8.23 ioadsels1

I/OxAPIC DSELS Register 1.

Type: CFG **Bus: 0** **Offset: 0x28c**	**PortID: N/A** **Device: 5**	**Function: 4**

Bit	Attr	Default	Description
17:0	RWS	0x0	gttcfg2SIpcIOADels1: gttcfg2SIpcIOADels1[17:0]

7.8.24 iointsrc0

I/O Interrupt Source Register 0.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2a0**				

Bit	Attr	Default	Description
31:0	RW-V	0x0	int_src0: bit interrupt source 31: INTD Port 3b 30: INTC Port 3b 29: INTB Port 3b 28: INTA Port 3b 27: INTD Port 3a 26: INTC Port 3a 25: INTB Port 3a 24: INTA Port 3a 23: INTD Port 1b 22: INTC Port 1b 21: INTB Port 1b 20: INTA Port 1b 19: INTD Port 1a 18: INTC Port 1a 17: INTB Port 1a 16: INTA Port 1a 15: INTD Port 2d 14: INTC Port 2d 13: INTB Port 2d 12: INTA Port 2d 11: INTD Port 2c 10: INTC Port 2c 9: INTB Port 2c 8: INTA Port 2c 7: INTD Port 2b 6: INTC Port 2b 5: INTB Port 2b 4: INTA Port 2b 3: INTD Port 2a 2: INTC Port 2a 1: INTB Port 2a 0: INTA Port 2a

7.8.25 iointsrc1

I/O Interrupt Source Register 1.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2a4**				

Bit	Attr	Default	Description
20:0	RW-V	0x0	int_src1: bit interrupt source 20: INTA Root Port Core 19: INTB ME KT 18: INTC ME IDE-R 17: INTD ME HECI 16: INTA ME HECI 15: INTD CB DMA 14: INTC CB DMA 13: INTB CB DMA 12: INTA CB DMA 11: INTD Port 0DMI 10: INTC Port 0DMI 9: INTB Port 0DMI 8: INTA Port 0DMI 7: INTD Port 3d 6: INTC Port 3d 5: INTB Port 3d 4: INTA Port 3d 3: INTD Port 3c 2: INTC Port 3c 1: INTB Port 3c 0: INTA Port 3c

7.8.26 ioremintcnt

Remote I/O Interrupt Count.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2a8**				

Bit	Attr	Default	Description
31:0	RW-V	0x0	rem_int_cnt: Number of remote interrupts received.

7.8.27 ioremgpecnt

Remote I/O GPE Count.

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2ac**				

Bit	Attr	Default	Description
23:16	RW-V	0x0	hpgpe_cnt: Number of remote HPGPEs received.
15:8	RW-V	0x0	pmgpe_cnt: Number of remote PMGPEs received.
7:0	RW-V	0x0	gpe_cnt: Number of remote GPEs received.

7.8.28 FauxGV

Type:	**CFG**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2c4**				

Bit	Attr	Default	Description
0:0	RWS_L	0x0	FauxGVEn: Enable Fault GV.

7.9 Device 5 Function 4 I/OxAPIC

I/OxAPIC has a direct memory mapped space. An index/data register pair is located within the directed memory mapped region and is used to access the redirection table entries. The offsets shown in the table are from the base address in either ABAR or MBAR or both.

Note: Access to addresses beyond 0x40h return all 0s.

Only addresses up to offset 0xFF can be accessed by means of the ABAR register whereas offsets up to 0xFFF can be accessed by means of MBAR.

Only aligned Dword reads and write are allowed towards the I/OxAPIC memory space. Any other accesses will result in an error.

Table 7-11. Integrated I/O Device 5 Function 4 I/O OxAPIC Register Address Map

Register Name	Offset	Size
index	0x0	8
window	0x10	32
eoi	0x40	8

7.9.1 index

The Index Register will select which indirect register appears in the window register to be manipulated by software. Software will program this register to select the desired APIC internal register.

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x0**				

Bit	Attr	Default	Description
7:0	RW-L	0x0	idx: Indirect register to access.

7.9.2 window

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x10**				

Bit	Attr	Default	Description
31:0	RW_LV	0x0	window_reg: Data to be written to the indirect registers on writes, and location of read data from the indirect register on reads.

7.9.3 eoi

Type:	**MEM**	**PortID:**	**8’h7e**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x40**				

Bit	Attr	Default	Description
7:0	RW-L	0x0	eoi_reg: The EOI register is present to provide a mechanism to efficiently convert level interrupts to edge triggered MSI interrupts. When a write is issued to this register, the I/O(x)APIC will check the lower 8 bits written to this register, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared. **Note:** If multiple I/O Redirection entries, for any reason, assign the same vector, each of those entries will have the Remote_IRR bit reset to ’0’. This will cause the corresponding I/OxAPIC entries to resample their level interrupt inputs and if they are still asserted, cause more MSI interrupt(s) (if unmasked) which will again set the Remote_IRR bit.

7.9.4 Device 5 Function 4 Window 0

Table 7-12. Integrated I/O Device 5 Function 4 Window 0 Register Address Map (Sheet 1 of 2)

Register Name	Offset	Size
apicid__window	0x0	32
ver__window	0x1	32
arbid__window	0x2	32
bcfg__window	0x3	32
rtl0__window	0x10	32
rth0__window	0x11	32
rtl1__window	0x12	32
rth1__window	0x13	32
rtl2__window	0x14	32
rth2__window	0x15	32
rtl3__window	0x16	32
rth3__window	0x17	32
rtl4__window	0x18	32
rth4__window	0x19	32
rtl5__window	0x1a	32
rth5__window	0x1b	32
rtl6__window	0x1c	32
rth6__window	0x1d	32
rtl7__window	0x1e	32
rth7__window	0x1f	32
rtl8__window	0x20	32
rth8__window	0x21	32
rtl9__window	0x22	32
rth9__window	0x23	32
rtl10__window	0x24	32
rth10__window	0x25	32
rtl11__window	0x26	32
rth11__window	0x27	32
rtl12__window	0x28	32
rth12__window	0x29	32
rtl13__window	0x2a	32
rth13__window	0x2b	32
rtl14__window	0x2c	32
rth14__window	0x2d	32
rtl15__window	0x2e	32
rth15__window	0x2f	32
rtl16__window	0x30	32
rth16__window	0x31	32

Table 7-12. Integrated I/O Device 5 Function 4 Window 0 Register Address Map (Sheet 2 of 2)

Register Name	Offset	Size
rtl17__window	0x32	32
rth17__window	0x33	32
rtl18__window	0x34	32
rth18__window	0x35	32
rtl19__window	0x36	32
rth19__window	0x37	32
rtl20__window	0x38	32
rth20__window	0x39	32
rtl21__window	0x3a	32
rth21__window	0x3b	32
rtl22__window	0x3c	32
rth22__window	0x3d	32
rtl23__window	0x3e	32
rth23__window	0x3f	32

7.9.4.1 apicid__window

This register uniquely identifies an APIC in the system. While this register is not used by operating systems anymore, it is still implemented in hardware.

Type: MEM **PortID:** N/A
Bus: 0 **Device:** 5 **Function:** 4
Offset: 0x0

Bit	Attr	Default	Description
27:24	RW	0x0	apicid: Allows for up to 16 unique APIC IDs in the system.

7.9.4.2 ver__window

This register uniquely identifies an APIC in the system. While this register is not used by operating systems anymore, it is still implemented in hardware.

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x1**				

Bit	Attr	Default	Description
23:16	RO	0x17	Maximum: This is the entry number of the highest entry in the redirection table. It is equal to the number of interrupt inputs minus one. This field is hardwired to 17h to indicate 24 interrupts.
15:15	RO	0x0	prq: This bit is set to 0 to indicate that this version of the I/OxAPIC does not implement the IRQ Assertion register and does not allow PCI devices to write to it to cause interrupts.
7:0	RO	0x20	vs: This identifies the implementation version. This field is hardwired to 20h indicate this is an I/OxAPIC.

7.9.4.3 arbid__window

This is a legacy register carried over from days of serial bus interrupt delivery. This register has no meaning in IIO. It just tracks the APICID register for compatibility reasons.

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x2**				

Bit	Attr	Default	Description
27:24	RO	0x0	arbitration_id: Just tracks the APICID register.

7.9.4.4 bcfg__window

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x3**				

Bit	Attr	Default	Description
0:0	RW	0x1	boot_configuration: This bit is a default1 to indicate FSB delivery mode. A value of 0 has no effect. Its left as RW for software compatibility reasons.

7.9.4.5 rtl[0:23]__window

The information in this register along with Redirection Table High Dword register is used to construct the MSI interrupt. There is one of these pairs of registers for every interrupt. The first interrupt has the redirection registers at offset 10h. The second interrupt at 12h, third at 14h, and so forth, until the final interrupt (interrupt 23) at 3Eh occurs.

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x10, 0x12, 0x14, 0x16, 0x18, 0x1a, 0x1c, 0x1e, 0x20, 0x22, 0x24, 0x26, 0x28, 0x2a, 0x2c, 0x2e, 0x30, 0x32, 0x34, 0x36, 0x38, 0x3a, 0x3c, 0x3e**				

Bit	Attr	Default	Description
17:17	RW	0x0	disable_flushing: This bit has no meaning in IIO. This bit is R/W for software compatibility reasons only
16:16	RW	0x1	msk: When cleared, an edge assertion or level (depending on bit 15 in this register) on the corresponding interrupt input results in delivery of an MSI interrupt using the contents of the corresponding redirection table high/low entry. When set, an edge or level on the corresponding interrupt input does not cause MSI Interrupts and no MSI interrupts are held pending as well (that is, if an edge interrupt asserted when the mask bit is set, no MSI interrupt is sent and the hardware does not remember the event to cause an MSI later when the mask is cleared). When set, assertion/deassertion of the corresponding interrupt input causes Assert/Deassert_INTx messages to be sent to the legacy ICH, provided the 'Disable PCI INTx Routing to ICH' bit is clear. If the latter is set, Assert/Deassert_INTx messages are not sent to the legacy ICH. When mask bit goes from 1 to 0 for an entry and the entry is programmed for level input, the input is sampled and if asserted, an MSI is sent. Also, if an Assert_INTx message was previously sent to the legacy ICH/internal-coalescing logic on behalf of the entry, when the mask bit is clear, then a Deassert_INTx event is scheduled on behalf of the entry (whether this event results in a Deassert_INTx message to the legacy ICH depends on whether there were other outstanding Deassert_INTx messages from other sources). When the mask bit goes from 0 to 1, and the corresponding interrupt input is already asserted, an Assert_INTx event is scheduled on behalf of the entry. **Note:** If the interrupt is deasserted when the bit transitions from 0 to 1, a Deassert_INTx is not scheduled on behalf of the entry.
15:15	RW	0x0	tm: This field indicates the type of signal on the interrupt input that triggers an interrupt. 0 indicates edge sensitive, 1 indicates level sensitive.
14:14	RO	0x0	rirr: This bit is used for level triggered interrupts; its meaning is undefined for edge triggered interrupts. For level triggered interrupts, this bit is set when an MSI interrupt has been issued by the I/OxAPIC into the system fabric (noting that if BME bit is clear or when the mask bit is set, no new MSI interrupts cannot be generated and this bit cannot transition from 0 to 1 in those conditions). It is reset (if set) when an EOI message is received from a local APIC with the appropriate vector number, at which time the level interrupt input corresponding to the entry is resampled causing one more MSI interrupt (if other enable bits are set) and causing this bit to be set again.
13:13	RW	0x0	ip: 0=active high; 1=active low. Strictly, speaking this bit has no meaning in IIO since the Assert/Deassert_INTx messages are level in-sensitive. But the core I/OxAPIC logic that is reused from PXH might be built to use this bit to determine the correct polarity. Most OS'es today support only active low interrupt inputs for PCI devices. Given that, the OS is expected to program a 1 into this register and so the 'internal' virtual wire signals in the IIO need to be active low that is, 0=asserted and 1=deasserted.

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x10, 0x12, 0x14, 0x16, 0x18, 0x1a, 0x1c, 0x1e, 0x20, 0x22, 0x24, 0x26, 0x28, 0x2a, 0x2c, 0x2e, 0x30, 0x32, 0x34, 0x36, 0x38, 0x3a, 0x3c, 0x3e**				

Bit	Attr	Default	Description
12:12	RO	0x0	delivery_status: When trigger mode is set to level and the entry is unmasked, this bit indicates the state of the level interrupt that is, 1b if interrupt is asserted else 0b. When the trigger mode is set to level but the entry is masked, this bit is always 0b. This bit is always 0b when trigger mode is set to edge.
11:11	RW	0x0	dstm: 0 - Physical1 - Logical
10:8	RW	0x0	delm: This field specifies how the APICs listed in the destination field should act upon reception of the interrupt. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. The encodings are:000 - Fixed: Trigger Mode can be edge or level. Examine TM bit to determine. 001 - Lowest Priority: Trigger Mode can be edge or level. Examine TM bit to determine. 010 - SMI/PMI: Trigger mode is always edge and TM bit is ignored. 011 - Reserved 100 - NMI. Trigger mode is always edge and TM bit is ignored. 101 - INIT. Trigger mode is always edge and TM bit is ignored. 110 - Reserved 111 - ExtINT. Trigger mode is always edge and TM bit is ignored.
7:0	RW	0x0	vct: This field contains the interrupt vector for this interrupt

7.9.4.6 rth[0:23]__window

Type:	**MEM**	**PortID:**	**N/A**		
Bus:	**0**	**Device:**	**5**	**Function:**	**4**
Offset:	**0x11, 0x13, 0x15, 0x17, 0x19, 0x1b, 0x1d, 0x1f, 0x21, 0x23, 0x25, 0x27, 0x29, 0x2b, 0x2d, 0x2f, 0x31, 0x33, 0x35, 0x37, 0x39, 0x3b, 0x3d, 0x3f**				

Bit	Attr	Default	Description
31:24	RW	0x0	did: They are bits [19:12] of the MSI address.
23:16	RW	0x0	edid: These bits become bits [11:4] of the MSI address.

§