Intel® Xeon Phi™ Coprocessor x200 Product Family

Datasheet

April 2017

Revision 001

Petitioner has added numbering to page 1. Otherwise, it leaves the original page numbering.

Document Number: 335808, Revision: 001US

MICROSOFT EXHIBIT 1033
Microsoft v. Partec
IPR2025-00318

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at Intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described 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.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

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.

Warning: Altering PC clock or memory frequency and/or voltage may (i) reduce system stability and use life of the system, memory 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 assumes no responsibility that the memory, included if used with altered clock frequencies and/or voltages, will be fit for any particular purpose. Check with memory manufacturer for warranty and additional details.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

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

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

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

Copyright © 2017, Intel Corporation. All Rights Reserved.

Revision History

Date	Revision	Description
April 2017	001	• Initial release

Table of Contents

1.0 Introduction 6
1.1 Reference Documentation 6
1.2 Terminology 6

2.0 Architecture 8
2.1 Board Overview 8
2.2 Board Placement 9
2.3 System Management Controller 10
2.4 Processor 11
2.5 Coprocessor Family 12

3.0 Thermal and Mechanical Specifications 13
3.1 Mechanical Specifications 13
3.1.1 System-Level Mechanical Retention 16
3.2 Thermal Specifications 16
3.2.1 Thermal Management 17
3.3 Thermal Solutions 17
3.3.1 Active Cooling Solution 17
3.3.2 Passive Cooling Solution 18

4.0 Electrical Specifications 21
4.1 PCI Express* Signals 21
4.1.1 PROCHOT_N Pin (Pin B12 or B30) 22
4.2 Supplemental Power Connectors 23

5.0 Power Management 24
5.1 Performance States (P-States), Power States (C-States), and Turbo Mode 24
5.2 System Sleep States (S-States) 25
5.3 Device Power States (D-States) 25
5.4 Link States (L-States) 26

6.0 Manageability 27
6.1 Coprocessor Manageability Architecture 27
6.2 System Management Controller (SMC) 27
6.3 General SMC Features and Capabilities 29
6.3.1 Catastrophic Shutdown Detection 29
6.4 Host/In-Band Management Interface (SCIF) 30
6.5 System and Power Management 31
6.5.1 IPMB Protocol 32
6.5.2 Polled Master-Only Protocol 32
6.5.3 Supported IPMI Commands 34
6.6 SMC LED_ERROR and Fan PWM 42

A Platform Reset Considerations 43

List of Figures

2-1 Coprocessor Block Diagram 8
2-2 Processor Block Diagram 9
2-3 Coprocessor Board, Front side 10
2-4 Coprocessor Board, Back side 10

2-5 Processor Tile Layout 11
3-1 Coprocessor Mounting Hole Locations 14
3-2 Coprocessor Product Dimensions 15
3-3 Hockey Stick Mechanical Retention Feature 16
3-4 Exploded View of the Active Solution 18
3-5 Exploded View of the Passive Solution 19
3-6 Airflow Requirement vs. Inlet Temperature for 300W Passive Coprocessor 20
5-1 Coprocessor P-States and Turbo 25
6-1 Coprocessor System Manageability Architecture 28
6-2 Write Block Command Diagram 33
6-3 Read Block Command Diagram 34
A-1 Topology Example of PERST# 43

List of Tables

1-1 Related Documents 6
1-2 General Terminology 6
2-1 Coprocessor Family SKUs 12
3-1 Coprocessor Mechanical Specifications 13
3-2 Coprocessor Thermal Specifications 16
4-1 PCI Express* Connector Signals on the Coprocessor 21
5-1 Coprocessor Power States 26
5-2 Supported Link States 26
6-1 SMBus Write Commands 33
6-2 Miscellaneous Command Details 34
6-3 FRU Related Command Details 34
6-4 SDR Related Command Details 35
6-5 SEL Related Command Details 35
6-6 Sensor Related Command Details 35
6-7 General Command Details 36
6-8 CPU Package Config Read Request Format 36
6-9 CPU Package Config Read Response Format 36
6-10 CPU Package Config Write Request Format 37
6-11 CPU Package Config Write Response Format 37
6-12 Set SM Signal Request Format 37
6-13 Set SM Signal Response Format 38
6-14 OEM Command Details 38
6-15 Set Fan PWM Adder Command Request Format 38
6-16 Set Fan PWM Adder Command Response Format 39
6-17 Get POST Register Request Format 39
6-18 Get POST Register Response Format 39
6-19 Assert Forced Throttle Request Format 39
6-20 Assert Forced Throttle Response Format 39
6-21 Enable External Throttle Request Format 40
6-22 Enable External Throttle Response Format 40
6-23 Table of Sensors 41
6-24 Status Sensor Report Format 42
6-25 LED Indicators 42

§

1.0 Introduction

The Intel® Xeon Phi™ Coprocessor x200 Product Family (formerly codenamed Knights Landing Coprocessors, and henceforth referred to as "coprocessor") is a series of PCI Express* 3.0 add-in cards containing an Intel® Xeon Phi™ Processor x200 Product Family processor in a BGA package (formerly codenamed Knights Landing Processor, and henceforth referred to as "processor") that allow the offload of massively-parallel algorithms used in High Performance Computing (HPC). This datasheet discusses the coprocessor architecture, mechanical, thermal, and electrical requirements, and the ways by which the host system can manage the card with respect to its temperature, power consumption, processor status, Field Replaceable Unit (FRU) information, etc.

1.1 Reference Documentation

Table 1-1 lists the documents referenced in this datasheet. Coprocessor drivers, utilities, applications, and other software for supported operating systems -- including helpful documentation such as the *Intel® Manycore Platform Software Stack (Intel® MPSS) User's Guide* -- is available on www.intel.com.

Table 1-1. Related Documents

Document	Source
Intel® Xeon Phi™ Coprocessor x200 Product Family Specification Update	www.intel.com
PCI Express Base Specification Revision 3.0*	www.pcisig.com
PCI Express Card Electromechanical Specification Revision 3.0*	www.pcisig.com
Advanced Configuration and Power Interface Specification	www.acpi.info
Intelligent Platform Management Bus Communications Protocol Specification, v1.0	www.intel.com
Intelligent Platform Management Interface Specification, v2.0	www.intel.com

1.2 Terminology

This section provides the definitions of terms used in the document.

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

Terminology	Definition
BGA	Ball Grid Array
BMC	Baseboard Management Controller
CFM	Cubic Feet per Minute
Coprocessor	Intel® Xeon Phi™ Processor x200 Product Family in the PCIe* add-in card form factor
CPU	Central Processing Unit
DMI	Direct Media Interface
DTS	Digital Thermal Sensor

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

Terminology	Definition
ECC	Error Correction Code
FET	Field Effect Transistor
I2C	Inter-IC bus
IHS	Integrated Heat Spreader
IPMB	Intelligent Platform Management Bus
IPMI	Intelligent Platform Management Interface
LFM	Low Frequency Mode
MCA	Machine Check Architecture
MCDRAM	Multi Channel Dynamic Random Access Memory
MCP	Multi-chip Package
ME	Manageability Engine
MIC	Many Integrated Core
Intel® MPSS	Intel® Manycore Platform Software Stack
NTB	Non-Transparent Bridge
PCB	Printed Circuit Board
PCH	Platform Controller Hub
PCIe*	PCI Express*
PECI	Platform Environment Control Interface
PID	Proportional Integral Derivative
PLD	Programmable Logic Device
RAS	Reliability Availability Serviceability
RHE	Remote Heat Exchanger
SCIF	Symmetric Communications Interface
SDR	Sensor Data Records
SEL	System Event Log
SKU	Stock Keeping Unit
SMBus	System Management Bus
SMC	System Management Controller
TDP	Thermal Design Power
TMTB	Thermal/Mechanical Test Board
VR	Voltage Regulator

§

2.0 Architecture

2.1 Board Overview

The coprocessor features:

- Processor with up to 68 Intel® architecture cores and 16 GB of high-bandwidth, on-package MCDRAM memory
- Intel® C610 Series Platform Controller Hub (PCH) with x4 DMI and PECI
- Non-Transparent Bridge (NTB) linking the processor to the edge connector
- x16 PCI Express* Gen3 interface with SMBus management interface
- SMC, thermal sensors, +12V power monitoring and on-board fan PID controller on the active SKU
- Coprocessor-level RAS features and recovery capabilities
- On-board flash device which configures the coprocessor after reset, and then loads a bootstrap which waits for the coprocessor OS image (pushed to it by Intel MPSS running on the host system)

Figure 2-1. Coprocessor Block Diagram

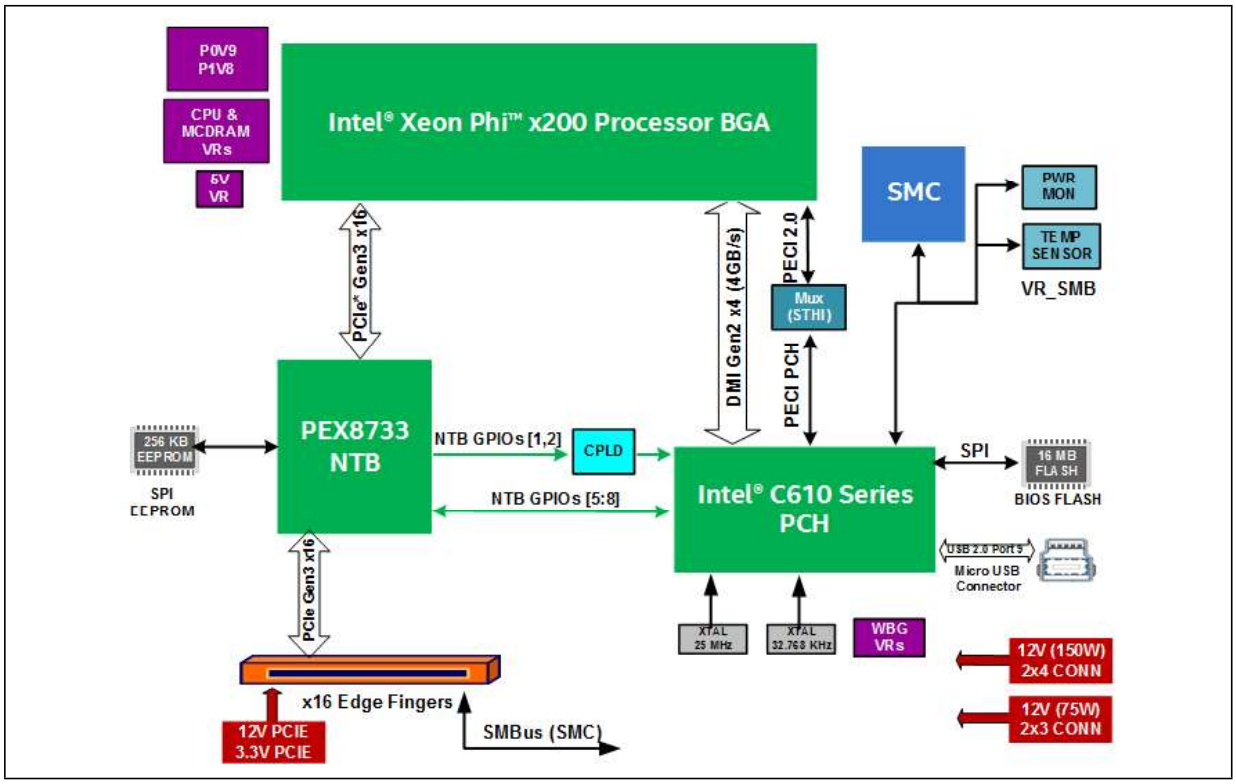
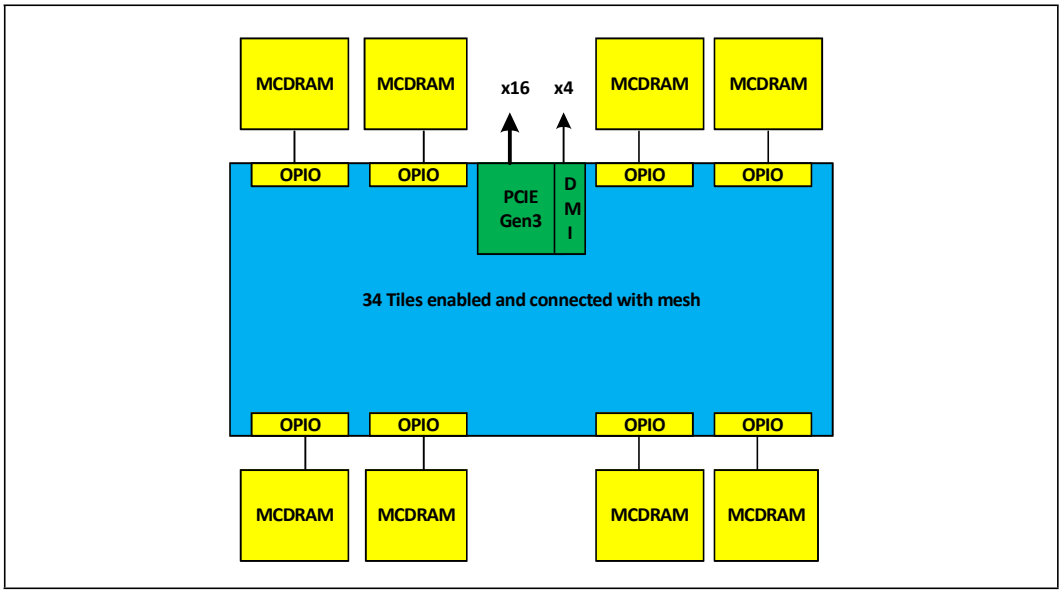


Figure 2-2. Processor Block Diagram



The processor includes:

- Multi-Chip Package (MCP) with CPU die and multiple stacked memory die (MCDRAM) under one Integrated Heat Spreader (IHS)
- x16 PCIe Gen3 interface
- DMI x4 connection to the PCH

2.2 Board Placement

The coprocessor is a PCIe 3.0 compliant high performance add-in card with an integrated thermal and mechanical solution. The processor includes 16 GB MCDRAM memory that provides over 500 GB/s effective bandwidth.

Figure 2-3 and Figure 2-4 show the front and back sides of the coprocessor PCB. The VR FETs and inductors are located east of the processor and on the west edge of the PCB. The PCH is located on the southeast area of the board, while the auxiliary power connectors are located on the east edge of the PCB. The NTB is located below the west VRs. There are 10 mounting holes in the PCB that retain the thermal solution and help apply preload to the processor.

Note: The figures below are a representation of the coprocessor board without the card's thermal/mechanical assembly. Component placement is subject to change.

Figure 2-3. Coprocessor Board, Front side

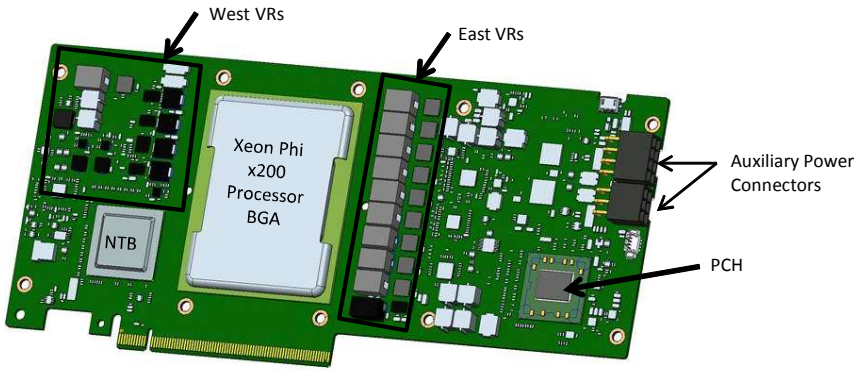
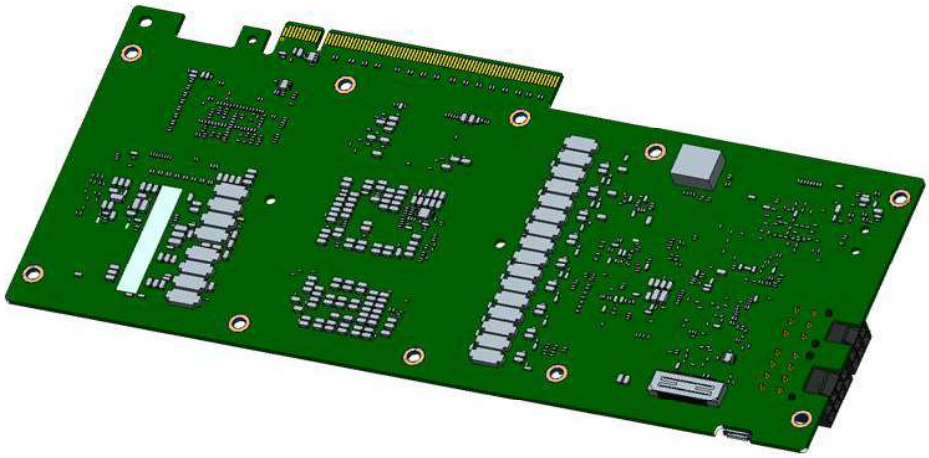


Figure 2-4. Coprocessor Board, Back side



2.3 System Management Controller

The on-board System Management Controller (SMC) has three I^2C interfaces. This allows a direct connection to the Intel C610 Series PCH I^2C interface, with the PCH in turn connecting to the processor via the DMI and PECI interfaces. The SMC/PCH interface is used for coprocessor thermal and status information. The sensor bus allows the board thermal, input power, and current sense monitoring for fan and power control. This information can be sent to the processor for power state control. An SMBus connection via the PCIe slot to the host platform can be used for system integration for chassis fan control on the passive heat sink coprocessor and integration with the Intel® Management Engine (Intel® ME) controller or Baseboard Management Controller (BMC) in the host platform. Refer to Section 3.2.1 for additional information on the coprocessor thermal management.

2.4 Processor

The processor is a BGA component attached to the board during assembly. See the following figure for a high-level architecture diagram.

Figure 2-5. Processor Tile Layout

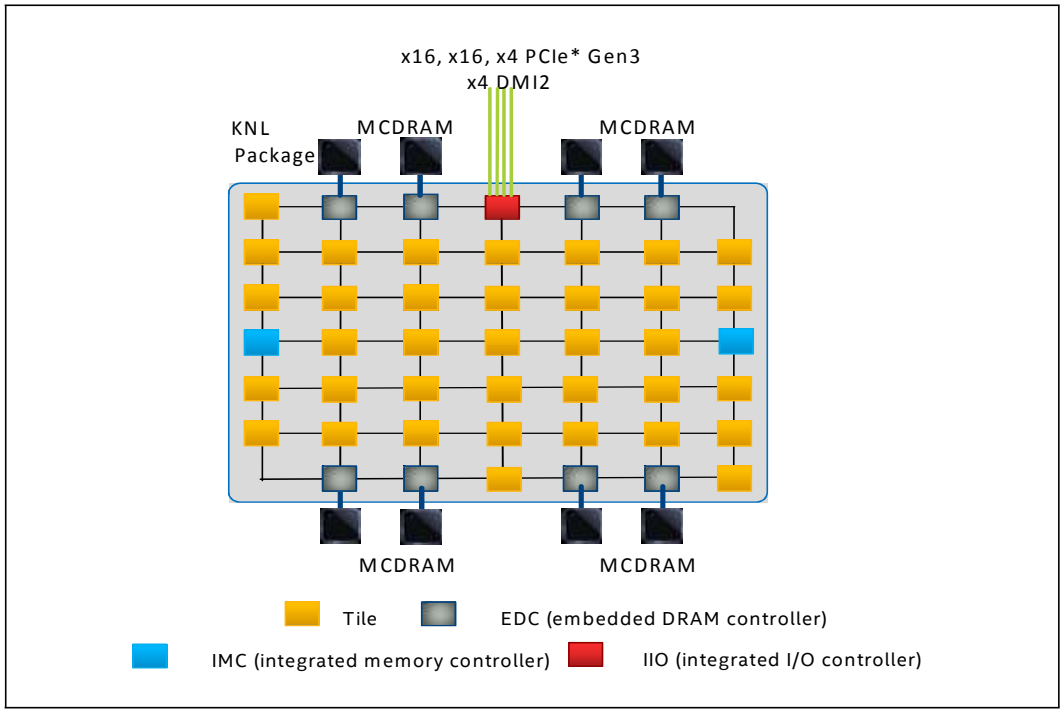


Figure 2-5 is a conceptual drawing of the general processor architecture, and does not imply actual distances or latencies. All tiles are connected via a two dimensional mesh. The high bandwidth interconnect mesh design efficiently allows each block on a row or column to be directly addressable by any other device on the same row or column. In the figure above, each yellow block is a tile containing two cores, shared L2 cache, and a mesh interface. The gray blocks are the on-package, high-speed stacked DRAM channels. The blue blocks are the DDR4 memory controllers, which are unused since there are no DDR4 modules on the card.

The L2 caches are located in each dual-core tile, but can also be thought of as a fully coherent cache, with a total size equal to the sum of the tiles. Information can be copied to each core as needed to provide the fastest possible local access, or a single copy can be present for all cores to provide maximum cache capacity.

The processor can support up to 68 cores (making a 34 MB L2 cache) and 16 GB of high-bandwidth MCDRAM. The core, mesh, and MCDRAM frequencies, and thermal solutions, vary by product SKU; refer to the *Intel® Xeon Phi™ Coprocessor x200 Product Family Specification Update* for more information.

2.5 Coprocessor Family

Table 2-1 lists the coprocessor family.

Table 2-1. Coprocessor Family SKUs

SKU	Coprocessor TDP (Watts)	Thermal Solution[1]	Hockey Stick Retention[2]
7240P	275	Passive	No
7220P	275	Passive	No
7220A	275	Active	No
7220A-HS	275	Active	Yes

Notes:
1. The passive thermal solution includes a topside heatsink and backplate with airflow required by the system. The active thermal solution has a topside heatsink and backplate and includes a dual-intake blower within the coprocessor thermal solution. See Section 3.3 for details.
2. See Section 3.1.1 for a further description of the mechanical board-based retention through the hockey stick.

§

3.0 Thermal and Mechanical Specifications

3.1 Mechanical Specifications

The mechanical features of the coprocessor are compliant with the *PCI Express* Card Electromechanical Specification Revision 3.0.*

The following table shows the mechanical specifications of all coprocessor products.

Table 3-1. Coprocessor Mechanical Specifications

Parameter	Specification
Product Length	246.9 mm[1]
Primary Side Height Keep-in	34.8 mm
Secondary Side Height Keep-in	2.67 mm
Active SKU Projected Mass	1270 g
Passive SKU Projected Mass	1245 g

Note: Inclusive of the I/O bracket, exclusive of th extender bracket.

Figure 3-1 shows the mounting holes and Figure 3-2 shows the relevant dimensions of the coprocessor for integration into the chassis. Both figures apply to all coprocessor products.

Figure 3-1. Coprocessor Mounting Hole Locations

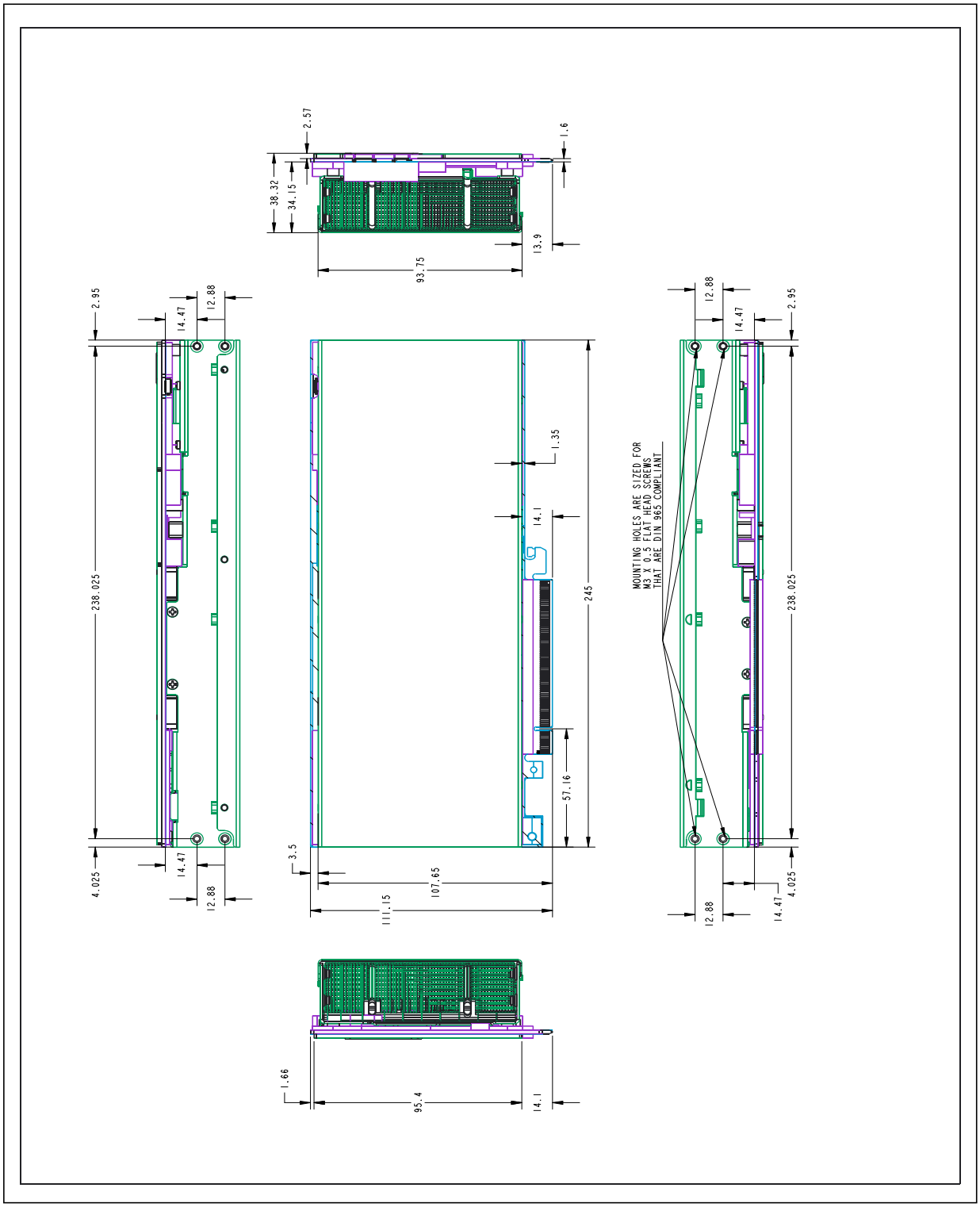
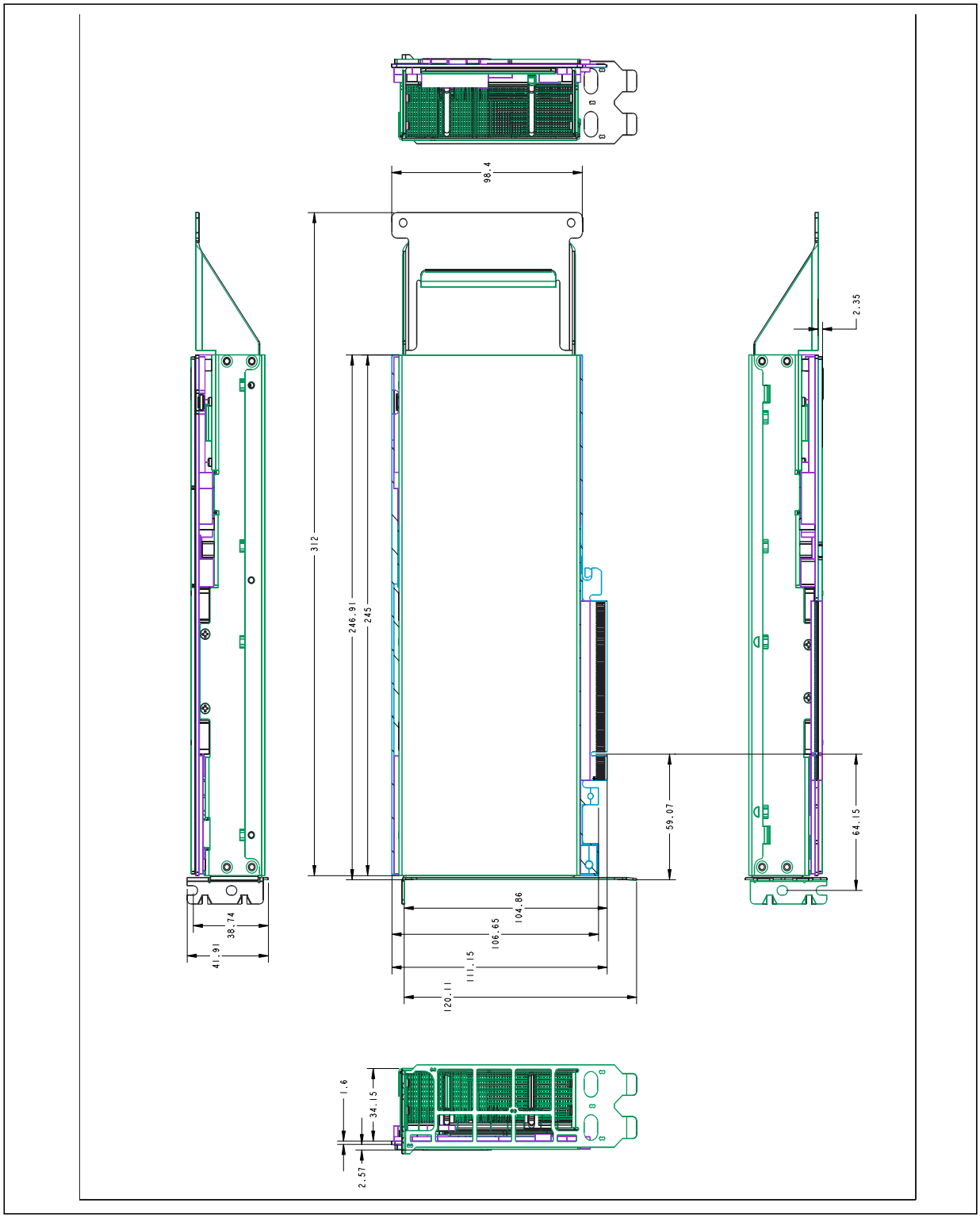


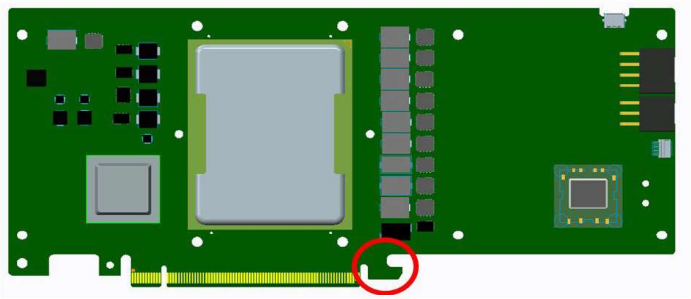
Figure 3-2. Coprocessor Product Dimensions



3.1.1 System-Level Mechanical Retention

Actively-cooled coprocessor SKUs include an optional retention mechanism intended for use in workstation systems. Sometimes referred to as a *hockey stick*, the mechanism is a PCB feature near the board edge fingers that engages into a system-provided latch when installed, as seen in Figure 3-3. This board-level retention mechanism protects the add-in coprocessor from electrical and mechanical damage during system shock and vibration events. See the *PCI Express* Card Electromechanical Specification Revision 3.0* for more information.

Figure 3-3. Hockey Stick Mechanical Retention Feature



3.2 Thermal Specifications

Table 3-2 lists the thermal specifications of the coprocessor.

Table 3-2. Coprocessor Thermal Specifications

Parameters	Specification
T_{RISE}	10 °C
Max T_{INLET}	45 °C
Max $T_{EXHAUST}$	70 °C
Card Tmin	5 °C
$T_{CONTROL}$ Range	90-94 °C[1]
$T_{THROTTLE}$	104 °C[2]
$T_{THERMTRIP}$	125 °C[3]

1. $T_{CONTROL}$ is the setpoint that should be maintained when the fan speed is less than 100%. This setpoint is programmed at the factory. Each processor is programmed at one $T_{CONTROL}$ value, but the value may be different from part to part.
2. $T_{THROTTLE}$ is the temperature at which the processor reduces its operating frequency in order to reduce power dissipation and allows the temperature to drop below the trip point.
3. If the processor temperature reaches $T_{THERMTRIP}$, the coprocessor takes action to shut down to prevent damage. This includes shutting down the coprocessor VRs. The only way to restart the coprocessor is by rebooting the host system. $T_{THERMTRIP}$ should not be considered a specification; it can change between SKUs and is given here for guidance only.
4. The information in this revision of the document is based on preliminary product characterization data. The values may change prior to production.

3.2.1 Thermal Management

Thermal management on the coprocessor is achieved through a combination of processor-based sensors and card-level sensors and inputs. The processor architecture includes thermal trip detection, automatic and adaptive thermal monitoring and PROCHOT_N signaling to reduce the processor temperature to protect its functionality and to optimize the its performance under a limited power budget.

The processor contains a factory-calibrated Digital Temperature Sensor (DTS) that monitors the CPU die temperature, also called the junction temperature or $T_{JUNCTION}$. Data from this sensor is available to the host BMC or other host system software via in-band (direct software reads) and out-of-band (over the PCI Express SMBus) methods. Refer to Chapter 6.0 for more information on how to read the junction temperature. System management software used by the host platform can use this data to monitor the CPU die temperature and take any appropriate actions. Systems that adjust airflow based on component temperatures must monitor the processor's DTS to ensure that sufficient cooling is always available, especially for the passive cards.

In addition to making thermal information available to the system manageability software, the DTS is constantly comparing the processor temperature to the factory-set maximum temperature called $T_{THROTTLE}$. If the measured temperature at any time exceeds $T_{THROTTLE}$, PROCHOT_N is asserted. Upon assertion, the processor automatically steps down to the lowest operating frequency or P-state, Pn, in an attempt to reduce the temperature. The thermal throttling continues until the temperature has dropped below $T_{THROTTLE}$, at which point the frequency is brought back up to the original setting.

3.3 Thermal Solutions

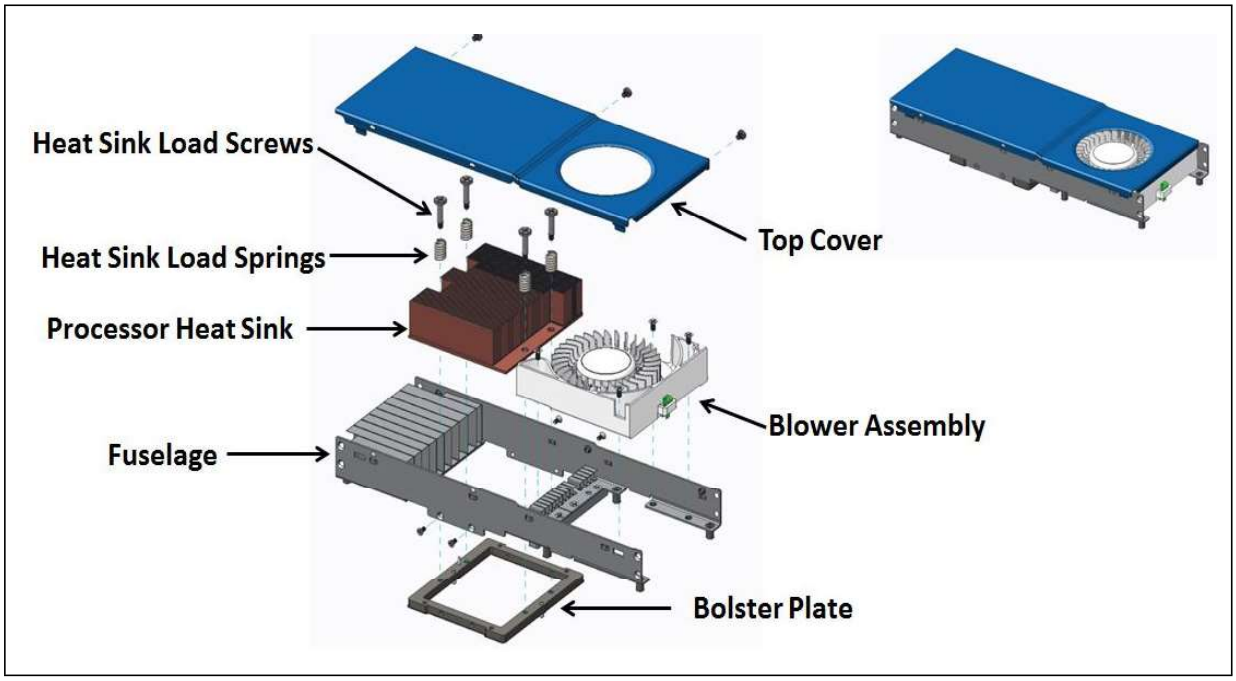
Active and passive thermal solutions are available to address the coprocessor power limits as indicated in Table 3-2. The passive thermal solution relies on forced convective airflow provided by the system, while the active thermal solution includes a high-performance blower to provide airflow to the coprocessor. The active solution is designed to operate in an *adjacent coprocessor configuration* such that the impedance from a nearby flow blockage is accounted for within the design. Both passive and active solutions have steel backplates that counteract the preload applied by the heatsink onto the processor. The backplate also protects the structural integrity of the coprocessor during a shock event.

Active and passive Thermal/Mechanical Test Boards (TMTB) replicate the thermal behavior and mechanical form of the coprocessor, and can be useful during host system development. Those interested in this tool should contact their Intel representative.

3.3.1 Active Cooling Solution

For the active design, the primary-side thermal-mechanical solution utilizes a sheet-metal aluminum *fuselage* housing with soldered fins to cool the VR components and provide structural integrity to the coprocessor. A high-performance blower provides the airflow needed to cool the thermally-significant components on the coprocessor. The primary-side thermal-mechanical solution is enclosed by a sheet-metal aluminum top cover that directs airflow through the coprocessor and provides structural rigidity. A step-down feature is also included on the cover to prevent the blower inlet from being blocked in dense, multiple-coprocessor configurations. The processor thermal path is separated and utilizes a heatsink with parallel copper plate fins and a copper base with two heat pipes. The heatsink is loaded by custom load springs. Figure 3-4 illustrates the key components of the active design.

Figure 3-4. Exploded View of the Active Solution

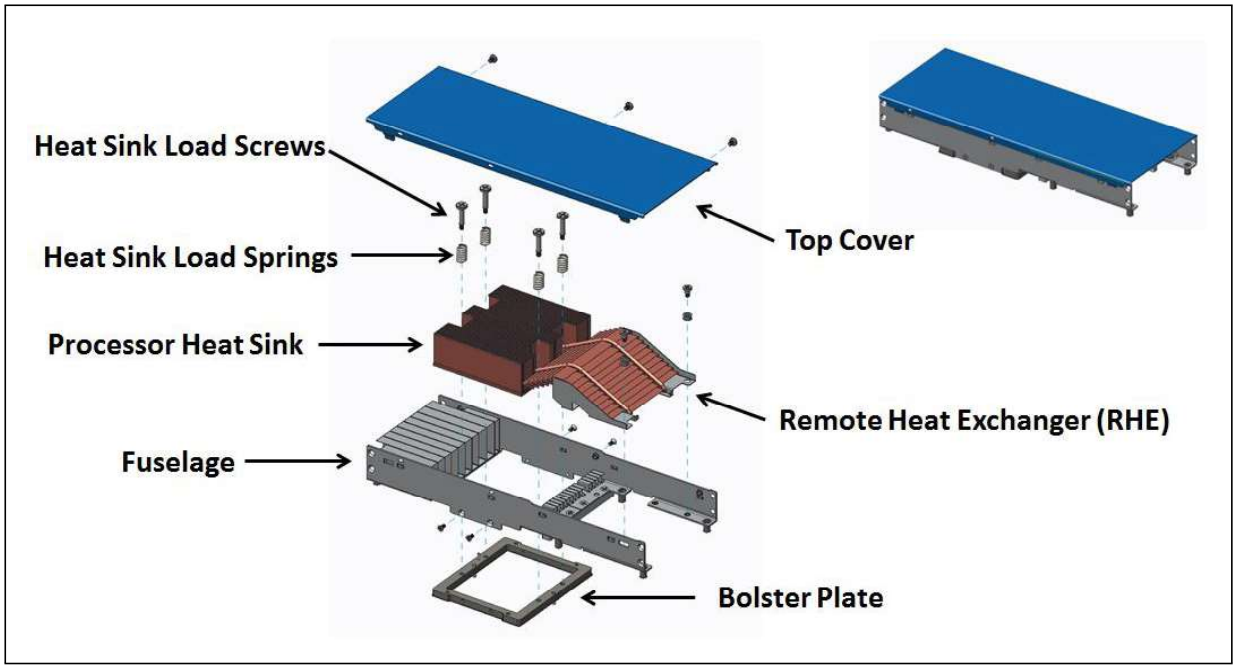


The active solution also contains a high-performance dual-intake blower that operates up to 6000 rpm at 20W of motor power. The blower is designed to maximize pressure-drop capability and can deliver up to 34.5 CFM with no adjacent blockage. When an adjacent coprocessor is considered, the resultant impedance loss causes the flow rate to drop to 28 CFM. The active thermal solution is designed to provide sufficient cooling even in the latter scenario.

3.3.2 Passive Cooling Solution

Similar to the active thermal-mechanical design, the passive thermal-mechanical solution has a sheet-metal aluminum *fuselage* housing the VR FET cooling fins and a processor heatsink. The processor heatsink is a separate component and thermal path in the thermal-mechanical assembly. The space that occupied the high-performance blower in the active thermal-mechanical assembly contains a Remote Heat Exchanger (RHE). The RHE contains heat pipes and copper fins soldered to an aluminum support bracket. Figure 3-5 illustrates the key components of the passive design.

Figure 3-5. Exploded View of the Passive Solution

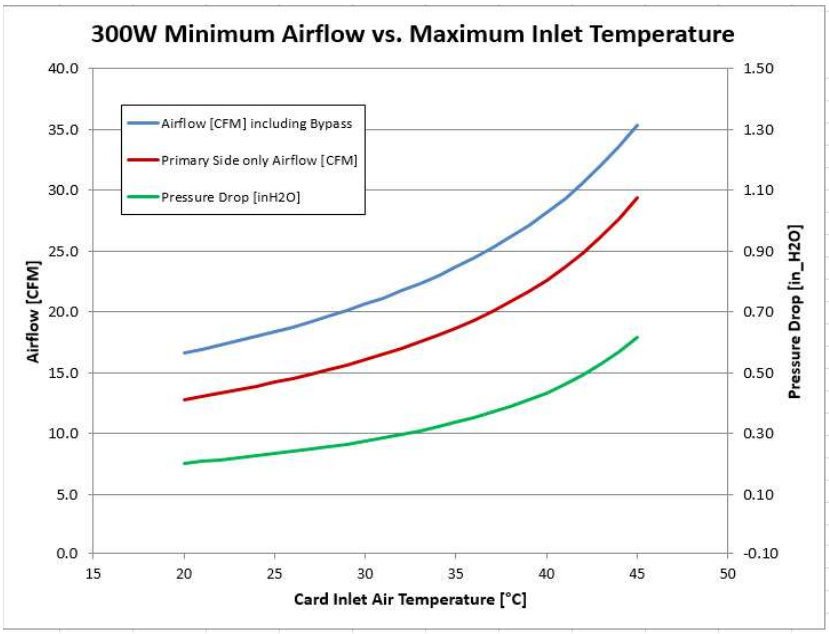


The passive solution requires forced convective airflow provided by the host system. Refer to the system airflow requirements, Section 3.3.2.1, for more information. For additional details on an open bench airflow test set-up, see the *PCI Express* Card Electromechanical Specification Revision 3.0.*

3.3.2.1 Passive Cooling Solution Airflow Requirements

To provide sufficient cooling of the passive coprocessor with a 45 °C inlet temperature, the system must be able to provide 35.4 CFM of airflow to the coprocessor as tested according to Section 10 of the *PCI Express* Card Electromechanical Specification Revision 3.0*. Total pressure drop assuming a multi-coprocessor installation conforming to the PCIe mechanical specification is 0.58 inH_2O at this flow rate. If the system is able to provide a temperature lower than 45 °C at the coprocessor inlet, then the total airflow can be reduced according to the graph and table in Figure 3-6.

Figure 3-6. Airflow Requirement vs. Inlet Temperature for 300W Passive Coprocessor



Card Inlet Temperature [°C]	Airflow [CFM] including Bypass	Primary Side only Airflow [CFM]	Pressure Drop [inH2O]
20	16.6	12.8	0.20
21	16.9	13.1	0.21
22	17.3	13.3	0.21
23	17.6	13.6	0.22
24	18.0	13.9	0.23
25	18.4	14.2	0.23
26	18.8	14.5	0.24
27	19.2	14.9	0.25
28	19.7	15.3	0.26
29	20.1	15.7	0.27
30	20.6	16.1	0.28
31	21.2	16.5	0.29
32	21.7	17.0	0.30
33	22.4	17.5	0.31
34	23.0	18.1	0.32
35	23.7	18.7	0.34
36	24.5	19.3	0.35
37	25.3	20.0	0.37
38	26.2	20.8	0.39
39	27.1	21.7	0.41
40	28.2	22.6	0.43
41	29.3	23.7	0.46
42	30.6	24.9	0.49
43	32.0	26.2	0.53
44	33.6	27.7	0.57
45	35.4	29.4	0.62

§

4.0 Electrical Specifications

4.1 PCI Express* Signals

The PCI Express connector for the coprocessor is a x16 Gen 3 interface and supports signals defined in the *PCI Express* Base Specification Revision 3.0*. Signals called out in the PCI Express specification but not used on the coprocessor are listed as "not used" in Table 4-1.

The symbol _N at the end of a signal name indicates that the active or asserted state occurs when the signal is at a low voltage level. When _N is not present after the signal name, the signal is asserted when at the high voltage level.

The following notations are used to describe the signal type:

I Signal is an Input to the coprocessor

O Signal is an Output from the coprocessor

I/O Bidirectional Input/Output signal

S Sense pin

P Power supply signal, sourced from the PCI Express edge fingers or supplemental power connectors.

Table 4-1. PCI Express* Connector Signals on the Coprocessor (Sheet 1 of 2)

Signal Name	Signal Type	Description
EXP_A_TX_[15:0]_DP EXP_A_TX_[15:0]_DN	O	PCI Express* Differential Transmit Pairs: 16-channel differential transmit pairs, referenced to the coprocessor. The EXP_A_TX_[15:0]_DP and EXP_A_TX_[15:0]_DN are connected to the PCI Express device transmit pairs on the coprocessor.
EXP_A_RX_[15:0]_DP EXP_A_RX_[15:0]_DN	I	PCI Express Differential Receive Pairs: 16-channel differential receive pairs referenced to the coprocessor. The EXP_A_RX_[15:0]_DP and EXP_A_RX_[15:0]_DN are connected to the PCI Express device receive pairs on the coprocessor.
CK_PE_100M_16PORT_DP CK_PE_100M_16PORT_DN	I	PCI Express Reference Clock: 100 MHz differential clock to the coprocessor, for it to properly recover data from the PCI Express interface.
PERST_N	I	PCI Express Reset Signal: PERST_N is a 3.3V active-low signal that when deasserted (high) indicates that the +12V and VCC3 power supplies are stable and within their specified tolerance.
SMB_PCI_CLK	I/O	PCI Express System Management Bus Clock: SMB_PCI_CLK is the 3.3V clock signal for the SMBus interface, which is normally used by the baseboard for power and/or thermal management and for monitoring the coprocessor.

Table 4-1. PCI Express* Connector Signals on the Coprocessor (Sheet 2 of 2)

Signal Name	Signal Type	Description
SMB_PCI_DAT	I/O	PCI Express System Management Bus Data: SMB_PCI_DAT is the 3.3V data signal for the SMBus Interface, which is normally used by the baseboard for power and/or thermal management and for monitoring the coprocessor.
PRSNT1_N, PRSNT2_N	S	Following PCI Express specification, PRSNT1_N (pin A1) is connected on the coprocessor card to PRSNT2_N (pin B81). Remaining PRSNT2_N pins (B17, B31, B48) are unconnected on the coprocessor card.
VCC3	P	+3.3V Supply: The positive 3.3V power supply to the PCI Express coprocessor
+12V	P	+12V Supply: The positive 12Vpower supply to the PCI Express coprocessor
V_3P3_PCIAUX	Not Used	+3.3V Auxiliary Supply
PROCHOT_N	I	Pin B12 and B30, both defined as reserved in the PCI Express specification, are user-defined options for throttling the coprocessor and for Intel® Xeon Phi™ coprocessor x100 family (formerly codenamed Knights Corner) backwards compatibility. It is driven by the baseboard and is pulled up to the 3.3V power rail on the coprocessor card when in the deasserted state. It must be driven active-low by the baseboard to exert throttling. See Section 4.1.1 and Chapter 6.0 for details.
WAKE_N	Not Used	PCI Express Wake Signal
EXP_JTAG[5:1]	Not Used	PCI Express JTAG Interface

4.1.1 PROCHOT_N Pin (Pin B12 or B30)

The coprocessor supports an external path through either the PCIe connector pin B12 or B30 to allow for the baseboard's system agents such as BMC or Intel ME to throttle the card, in response to system thermal events. The SMC through IPMI commands enables/disables this functionality as well as selects which pin is used. See Section 6.5.3.7.4.

Users may choose to enable pin B12 instead for backwards compatibility with the Intel® Xeon Phi™ coprocessor x100 systems.

System baseboard routing to the PROCHOT_N pin must take into consideration the following details:

- The PROCHOT_N pin is driven by the +3.3V power rail.
- The PROCHOT_N pin is connected to a pull-up of 1 kΩ on the coprocessor.
- The input signal arriving at the pin from the baseboard must meet the following characteristics:
  - $V_{IH(min)}$= 2.7V
  - $V_{IL(max)}$= 0.5V
  - Rise/fall times(max)= 240 ns
- The baseboard implementation can choose to be either push-pull or open-drain.

Note: Some PROCHOT_N events are not masked when resetting the coprocessor, causing it to fail to start after the reset sequence (i.e., PROCHOT_N must be deasserted prior to beginning the coprocessor reset sequence):

- The PROCHOT_N pin on the PCIe connector (if enabled)
- OEM Assert Forced Throttle (see Section 6.5.3.7.3)

4.2 Supplemental Power Connectors

The coprocessor gets a maximum of 75W from the PCI Express connector, per the PCI Express specification. The 2x4 and 2x3 supplemental power connectors on the coprocessor provide the additional +12V power needed by the coprocessor. Per the PCI Express specification, the 2x4 connector must be capable of a maximum 150W power draw by the coprocessor, and the 2x3 must be capable of a maximum 75W power draw. All SKUs of the coprocessor family must have power supplied to both the 2x4 and the 2x3 connectors. Within the coprocessor, the power rails from the three sources are not connected to each other. Instead, the coprocessor is designed to draw power proportionally from the three power sources. During coprocessor power-up, sensors on the coprocessor detect the presence of power supplies on the supplemental connectors, and can determine if sufficient power is available to power up the coprocessor. For the coprocessor, the sensors must detect both 2x4 and 2x3 power supplies in order for the coprocessor to be powered up and function properly. A 12V Over Voltage/Under Voltage (OV/UV) protection is implemented by the usage of the internal VR12.0 controller OV/UV.

§

5.0 Power Management

Power management consists of monitoring and managing power dissipation on the coprocessor; it is primarily performed by the on-card resident coprocessor OS with hardware-controlled functionality. The host platform can typically view, set, and use the coprocessor power dissipation as a part of an overall power management scheme. The coprocessor supports various power, device, link, and performance states as described below.

5.1 Performance States (P-States), Power States (C-States), and Turbo Mode

P-states, or Performance states, are different frequency settings requested by the pCode when the cores are in the C0 active/executing state. Switching between P-states is done by the coprocessor when the OS or application determines that more or less performance is needed. All active cores run at the same P-state frequency as there is only one clock source in the coprocessor.

Each frequency setting of the coprocessor requires a specific Voltage Identification (VID) voltage setting in order to guarantee proper operation, and each P-state corresponds to one of these frequency and voltage pairs. Each device is uniquely calibrated and programmed at the factory with its appropriate frequency and voltage pairs. As a result, it is possible that two devices of the same SKU may have different voltage settings.

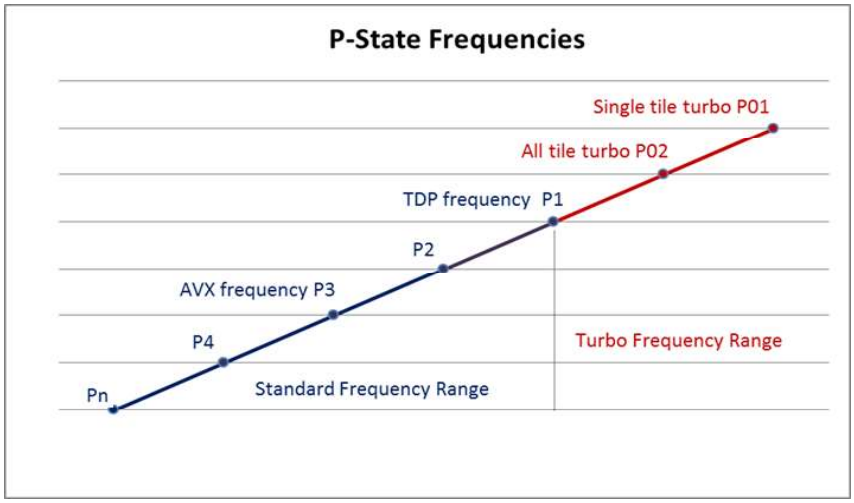
The coprocessor supports Intel® Turbo Boost Technology which opportunistically runs the coprocessor at a higher frequency than its TDP rated value. When the coprocessor is operating below its specified power and temperature limits, the Power Control Unit (PCU) within the coprocessor selects the highest possible turbo frequency while still remaining within the power and thermal specifications.

The highest turbo mode P-state is P01 which is single-tile turbo, followed by the all-tile turbo state of P02. P-states within the standard frequency range are referred to as P1 (non-AVX TDP frequency), P2, P3 (AVX frequency), P4 and Pn being the lowest frequency state. Pn, also called LFM or Low Frequency Mode, is only used by the coprocessor when the device is over $T_{THROTTLE}$ and is attempting to cool down by reducing power dissipation. See Figure 5-1. All parts within a given SKU have the same P-states, but P-states and turbo frequencies may vary across SKUs.

Once the OS requests turbo operation by selecting the P01 state, the coprocessor automatically selects the best P0n state that remains within the specified thermal and power limits. Determination of this P-state is based on the number of active cores, the current draw, the average power consumption, and the temperature. If these conditions change, the turbo P-state may also change or even be reduced to the non-turbo P-state of P1. In turbo mode, the coprocessor is free to change the P-state at any time without giving advanced notice to the OS. Although the OS may request P01, there is no guarantee that a turbo frequency is selected. If the conditions are not sufficient to allow the coprocessor to run above P1, then it remains in P1. The amount of time the processor can spend in turbo mode may be influenced by the workload and the operating environment.

Turbo mode may be disabled through the SMC Control Panel, or by configuring the operating system such that it never requests the P01 P-state.

Figure 5-1. Coprocessor P-States and Turbo



5.2 System Sleep States (S-States)

The coprocessor supports S0 device wake/active on and S5 software off states. Since the coprocessor relies on a host system for both functionality and power, it has a unique S5 implementation.

Upon S5 entry, the coprocessor BIOS sends the coprocessor into the C6 state, which halts all the threads. Keeping the threads in C6 allows the coprocessor to enter a deep package C-state and minimize power consumption. The Intel ME or host software resets the coprocessor to transition from S5 to S0.

5.3 Device Power States (D-States)

When a coprocessor is installed into a system, the Non-Transparent Bridge (NTB) communicates between the processor endpoint and host system endpoint via the PCIe bus. The coprocessor supports the D0 (fully-on) and $D3_{cold}$ (off) states.

Note: The coprocessor does not support the $D3_{hot}$ state.

For additional information about device states, refer to the *Advanced Configuration and Power Interface Specification.*

Table 5-1 shows estimates for coprocessor power and wakeup times for D-state and S-state combinations.

Table 5-1. Coprocessor Power States

Coprocessor Power State	Processor Power	Total Coprocessor Power	Snoop Latency	Exit Latency/ Wakeup Time
S0 (active on)/D0 (on)				
Package C0	205W	275W	N/A	N/A
Package C6	~15W	~38W	~40 µs	<140 µs
S5 (software off)/$D3_{cold}$ (off)	0W	~10W (Passive SKU) ~12W (Active SKU)	N/A	N/A

5.4 Link States (L-States)

The majority of PCIe power consumption is due to lane activities. The coprocessor supports L0 (active on) and L3 (off) states. See Table for more information.

Dynamic Lane Width (DLW) reduces power consumption over the PCIe lanes by reducing the number active lanes to the required bandwidth needed. The coprocessor does not support DLW.

Table 5-2. Supported Link States

L-State	Corresponding Package C-State
L0	Package C0/C2/C3/C6
L3	Disable lanes

§

6.0 Manageability

6.1 Coprocessor Manageability Architecture

The server management and control panel component of the coprocessor architecture provides a system administrator with the runtime status of the coprocessor installed in a given system. There are two access methods by which the server management and control panel component may obtain status information from the coprocessor. The *in-band* method utilizes the Symmetric Communications Interface (SCIF) network and the capabilities designed into the coprocessor OS and the host driver to deliver the coprocessor status. It also provides a limited ability to set specific parameters that control hardware behavior. The same information can be obtained using the *out-of-band* method. This method starts with the same capabilities in the coprocessor OS, but sends the information to the System Management Controller (SMC) using a proprietary protocol. The SMC responds to queries from the host platform's Baseboard Management Controller (BMC) using the Intelligent Platform Management Interface (IPMI) protocol to pass the information upstream to the administrator or user. IPMI to the SMC is available even when the coprocessor is idle, providing a route for reading idle power.

6.2 System Management Controller (SMC)

Coprocessor manageability relies on an SMC. The system provides sensor telemetry information for management by in-band (host) software and out-of-band software via the PCI Express SMBus. The SMC also provides additional functionality as described in this chapter.

The SMC is a micro-controller-based thermal management and communications system that provides coprocessor-level control and monitoring. Thermal management is achieved through monitoring the processor and the various temperature sensors located on the coprocessor card. Coprocessor card level power management monitors the coprocessor input power and communicates current power conditions to the processor.

SMC features include:

- Multiple thermal sensor level inputs: CPU die, VR area (board), processor area (board), NTB area (board), airflow out area (board), etc
- Power alert, thermal throttle, coprocessor-level power limiting, power/energy measurement, fan data access for active-cooled SKUs, and THERMTRIP_N signals

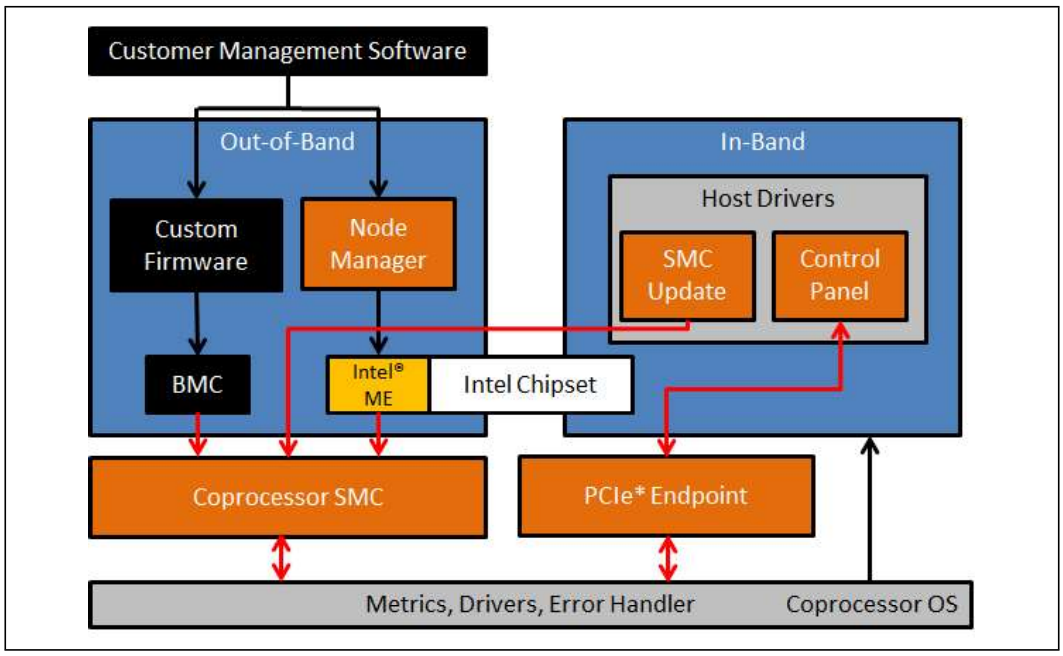
The SMC connects the host to the processor via in-band and out-of-band signals:

- In-band communication
  - gmond exposed via the standard Ethernet port
  - Accessible via the control panel GUI and API
- Out-of-band communication
  - Access to the SMC via the PCI Express SMBus using the IPMB protocol

— 50 ms sampling rate for power data

The manageability architecture also provides support for the coprocessor via the Intel® Node Manager (Intel® NM), which adds functionality such as setting power throttle threshold values tracked over separate time windows, and power and thermal assist. If a BMC is present, then it supports power up, MCTP, and BMC assist.

Figure 6-1. Coprocessor System Manageability Architecture



In operational mode, the SMC monitors power and temperatures within the processor and through sensors located on the PCI Express coprocessor board. This information is then used to control the power consumed by the PCI Express coprocessor and the rotating speed of the fan(s) within the PCI Express coprocessor cooling system. The SMC provides status information (temperature, fan speed, and voltage levels) to the coprocessor drivers, which then can be provided to the end user via a GUI. The SMC provides a master/slave SMBus (using the IPMB protocol) so that a platform BMC or Intel ME can control the SMC.

The SMC on the coprocessor has the following capabilities:

- General manageability features
- Board ID and SKU definition
- Unique identifying number
- Fan control
  - — Read fan RPM
- Thermal throttling and throttle monitoring
  - — Force throttling of the coprocessor
  - — Emergency throttling via pin B12 or B30 of the PCIe connector
  - — Monitor time in throttled state
  - — Separated status if power throttling vs. over-temperature throttling
- Coprocessor-level power throttle threshold/capping

  - Power throttle threshold values 0 and 1, tracked over separate time windows
  - P-state clamping if the P-state requested is not possible within the set power envelope
- Power/energy measurement
  - Can choose to include or preclude 3.3V power

6.3 General SMC Features and Capabilities

The coprocessor supports the PCI Express 3.0 standard. The SMC located on the coprocessor has direct access to information about the coprocessor operation (such as fan speeds, power usage, etc.) that may be managed from host-based software.

The SMC supports manageability interfaces via the SCIF interface which is part of the Intel® Manycore Platform Software Stack (Intel® MPSS) software stack and the preferred PCI Express SMBus (IPMB protocol) as well as with polled master only IPMI protocol.

The SMC firmware update process is resilient against unexpected power loss and resets.

The SMC supports a read-only, IPMI-compliant Field Replaceable Unit (FRU) that contains the following information:

- Manufacturer name
- Product name
- Part number/model number
- Universal Unique Identifier (UUID)
- Manufacturer's IPMI ID
- Product IPMI ID
- Manufacturing time/date stamp
- Serial number (12 ASCII bytes)

On SKUs with active cooling, to keep the coprocessor within the operational temperature range, the SMC will adjust the integrated fan's speed between 50% and 100%. During a thermal event where THERMTRIP_N asserts, the SMC will boost the fan to full speed. On SKUs with passive cooling solutions, the host system is responsible for providing proper cooling, as specified in Section 3.3.2.1.

Additionally the SMC supports enabling and disabling an external assertion path from the baseboard to the coprocessor via pin B12 or B30 depending on the host system configuration. This allows an external agent, such as a BMC or Intel ME, to force throttle the coprocessor during thermal events. See Section 4.1.1 for baseboard implementation details.

6.3.1 Catastrophic Shutdown Detection

Catastrophic shutdown is the act of the coprocessor shutting itself down to prevent damage to the device caused by overheating. The SMC detects this event by monitoring the THERMTRIP_N (or a modified version of the same signal). When THERMTRIP_N is asserted (low), the SMC will detect this event, immediately force the fan(s) to full speed. Either a soft reset or removal of power is required to reset the microcontroller to a known start point.

6.4 Host/In-Band Management Interface (SCIF)

Manageability, through the SMC, is achievable via the SCIF interface which is part of the Intel® MPSS software stack. This allows host programs to obtain telemetry and other information from the SMC-managed features of the coprocessor itself as well as control SMC-enabled functions.

The following SMC information and sensors are accessible over the host-based user-mode SCIF interface:

- Hardware strapping pins
- SMC firmware revision number
- UUID
- PCI-compliant Memory Mapped Input/Output (MMIO)
- Fan tachometer
- Fan Pulse-Width-Modulation (PWM) to boost fan speed for additional cooling
- SMC System Event Log (SEL)
- Voltage rail discrete monitoring
- All discrete temperature sensors
- $T_{CRITICAL}$
- $T_{CONTROL}$
- $T_{CURRENT}$
- Thermal throttle duration due to coprocessor power throttle threshold (in ms), free running counter that overflows at 60 seconds
- T_{INLET_TEMP} (derived number)
- T_{EXHST_TEMP} (derived number)
- PERF_Status_Thermal
- 32-bit POST register
- SMC System Event Log (SEL) entry select and data registers (read only)
- SMC Sensor Data Records (SDR) entry select and data registers (read only - required to interpret the SEL)

Each SMC sensor that is exposed over SCIF indicates one of four states in a consistent manner, returned in the same register as the sensor reading itself, regardless of sensor type. These states do not apply to non-sensor information:

- Normal
- Upper critical
- Lower critical
- Inaccessible (sensor not available)

This minimizes the complexity of host-driven software and SMC firmware implementations.

The sensors available from the SMC vary within the Intel® Xeon Phi™ coprocessor family of products. However, the IPMI SDR sensor names do not change from release to release. Therefore, one should search by sensor name, not sensor number, in order to be backwards compatible when using scripts to search for sensors.

T_{INLET_TEMP} and T_{EXHST_TEMP} are derived numbers and keep a running record of the lowest and highest temperature values, respectively, of all the board temperature sensors.

The SMC assigns multiple thermal sensors on the coprocessor: in the CPU die, on the board near the processor, near the NTB, near the VRs, near the airflow out area, and so forth. It is important to note that the "inlet" and "exhaust" thermal sensors are not airflow temperature sensors.

There are also power sensors attached to the 12V power inputs from the PCIe slot, the 2x3 connector, and the 2x4 connector. Input power can be estimated by summing the currents over these three connections. For an actively-cooled coprocessor, the SMC can also provide the fan percentage PWM being used. Fan speed is a simple Proportional Integral Derivative (PID) control algorithm with setpoints set rather high to keep the sound level low when maximum cooling is not needed.

6.5 System and Power Management

The coprocessor supports both on-coprocessor power management and an option for system-based management. With on-coprocessor power management, the SMC adjusts coprocessor power using preprogrammed power throttle threshold values. With system-based management, the SMC receives power control inputs via in-band communication from a host application or out-of-band via IPMB commands from a host BMC.

The coprocessor exists as part of a system. For this system to manage its cooling and power demands, the coprocessor telemetry must be exposed to ensure that the system is adequately cooled and that proper power is maintained. Manageability code running elsewhere in the chassis, through the SMC, can retrieve SMC sensor logs, sensor data, and vital information required for robust server management. Note that logging, in this context, is completely separate from and has nothing to do with the Machine Check Architecture (MCA) error log.

The SMC public interface (SMBus) is a compliant IPMB interface. It supports a minimal IPMB command set in order to interact with manageability devices on the baseboard such as BMCs and the Intel ME.

The IPMB implementation on the SMC can receive additional incoming requests while responses are being processed. This enables the interleaving of requests and responses from multiple sources using the SMC's IPMB, thus minimizing latency.

Upon initial power-on or restart, the SMC selects an IPMB slave address from the range 0x30 - 0x4e in increments of 2 (for example, 0x30, 0x32, 0x34, and so forth). The IPMB slave address self-select starting address is nonvolatile, starting at the last selected slave address. This ensures that the coprocessor address does not move nondeterministically in a static system. To determine the address of the coprocessor, scan the range of addresses issuing the Get Device ID command for each address. A valid response indicates the address used is a valid address.

For the coprocessor, the IPMB slave address may be found at 0x32 if only a single coprocessor is installed. If the motherboard has an exclusive connection to the SMBus on each PCI Express connection, then the coprocessor assigns itself a default address (0x30). If the SMBus connections are shared, each coprocessor in a chassis negotiates with each other and selects addresses in the range from 0x30 to 0x4e. If a mux is incorporated into the design to isolate devices on a shared link, the address negotiation process should result in each coprocessor having address 0x30. However, if the mux in use allows for the channels to be merged, i.e., creating a shared bus scenario, the address negotiation may result in each coprocessor having a unique address behind the mux. Due to factors such as noise traffic on the PCIe SMBus, the address of 0x30 is not guaranteed, and an address in the range of 0x30 to 0x4e may be selected.

Power management and power control are performed through the host driver interface (in-band). An SDK is provided as part of the coprocessor software stack and can be found in the standard Intel MPSS release.

The SMC's PCI Express/SMBus interface operates as an industry standard IPMB with a reduced IPMI command implementation. The SMC supports a System Event Log (SEL) via the IPMI interface.

The SMC supports a read-only IPMI Sensor Data Records (SDR). It is hard-coded and cannot be updated by end users. The SDR can be read in "chunks" (suggested size is 16 bytes) or the entire SDR can be read (passing 'FF' as the number of bytes to read).

6.5.1 IPMB Protocol

The IPMB protocol is a symmetrical byte-level transport for transferring IPMI messages between intelligent I^2C devices. It is a worldwide standard widely used in the server management industry. In this case, the host requests are sent to the SMC with a master I^2C write.

Although both devices are masters on the bus at different times, the SMC only responds to requests. With the exception of the address selection algorithm, it does not initiate master transactions on the bus at any other time during normal operation.

The commands supported by the SMC are documented below. The specific information to implement these commands is documented with each command. For byte level details, refer to the *Intelligent Platform Management Bus Communications Protocol Specification, v1.0* and the *Intelligent Platform Management Interface Specification, v2.0*.

6.5.2 Polled Master-Only Protocol

The polled master-only protocol may be used in the event IPMB is not feasible. The host sends requests to the SMC using one or more SMC SMBus write block commands then, at a later time, reads the response using one or more SMBus read block commands.

6.5.2.1 Polled Master-Only Protocol Clarifications

The polled master-only protocol is loosely based on the IPMI defined SSIF protocol; however, there have been a few changes made and ambiguities clarified to make the protocol more reliable:

- The I^2C address for the polled master-only protocol and the IPMB protocol are the same and work together transparently.
- PEC bytes are required for all write commands and are returned with all valid read responses.
- The maximum SMBus data length is restricted to 32 bytes.
- The SMC ignores write commands that occur while it is internally processing a previous command.
- The SMC does not return valid data while busy internally processing a command.
- A sequence number has been added to help identify the condition where a new write command (using the same NetFn and command as the last command sent) was corrupted during transit. Without this precaution, two sequential requests of the same type (i.e., Get Sensor Reading) could result in one sensor's reading being mistaken for the other's reading.
- SMBAlert is not supported.

6.5.2.2 SMBus Write and Read Block Command Numbers

6.5.2.3 Read and Write Descriptions

Table 6-1. SMBus Write Commands

Command	Name	Command Type
02h	Single Part Write	Write Block
06h	Multi-Part Write Start	Write Block
07h	Multi-Part Write Middle	Write Block
08h	Multi-Part Write End	Write Block
03h	Single Read Start	Read Block
03h	Multi-Part Read Start	Read Block
09h	Multi-Part Read Middle	Read Block
09h	Multi-Part Read End	Read Block

Figure 6-2. Write Block Command Diagram

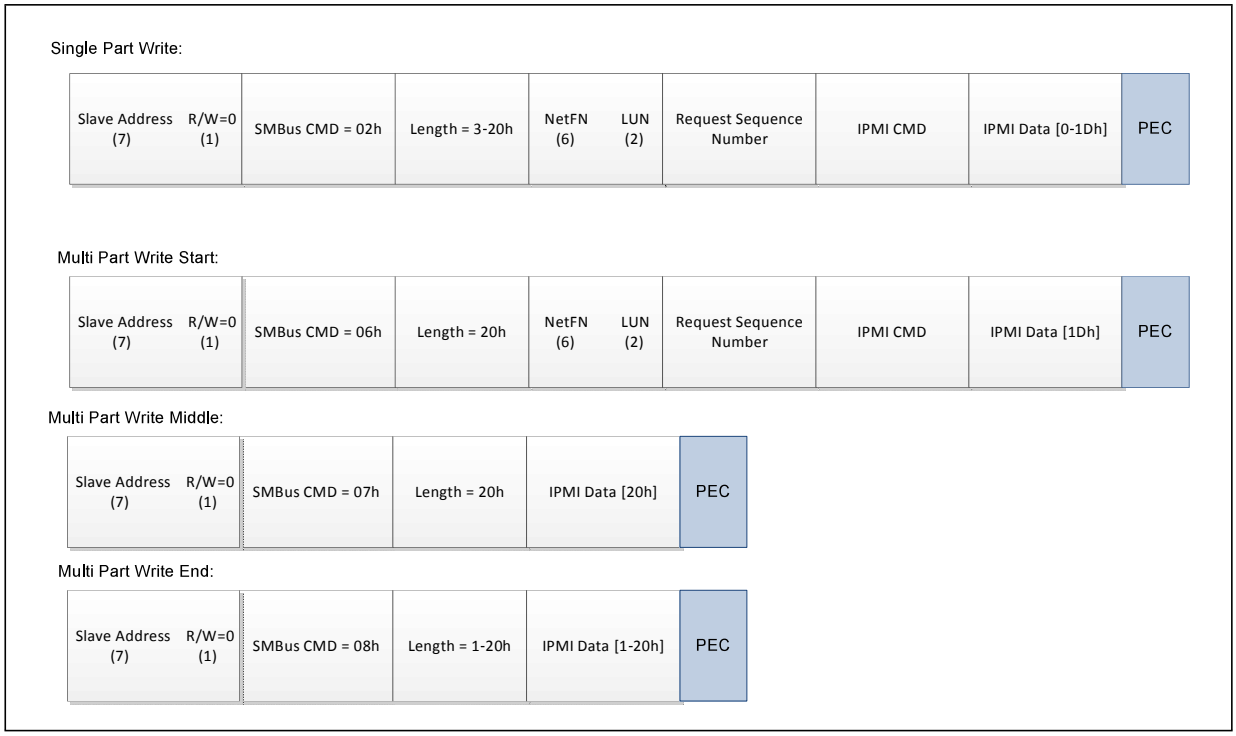
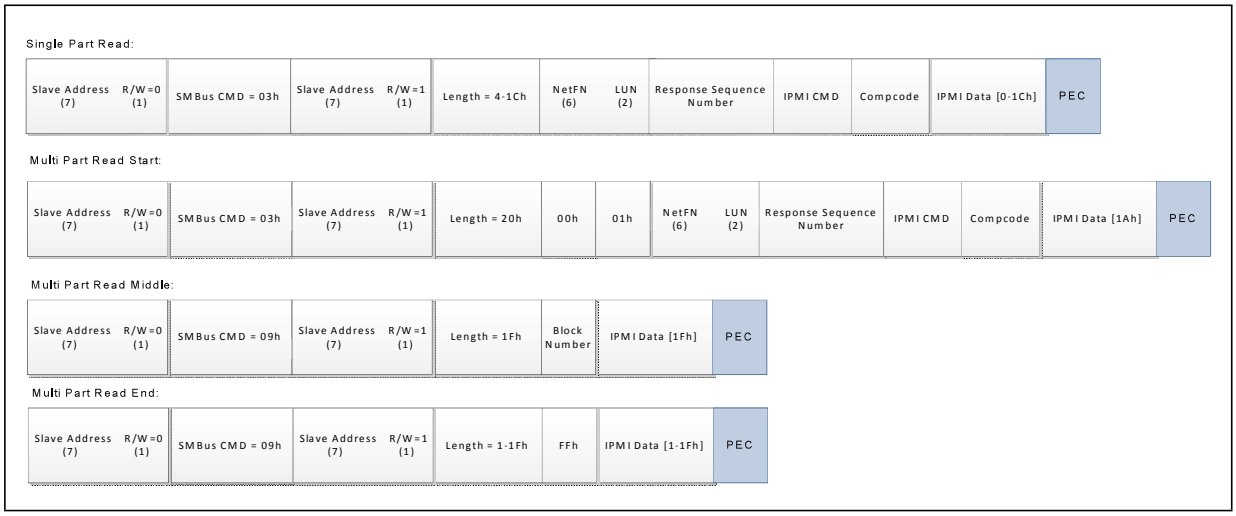


Figure 6-3. Read Block Command Diagram



6.5.3 Supported IPMI Commands

The SMC supports a subset of the standard IPMI sensor, SEL, and SDR commands along with several Intel OEM commands for accomplishing things like forcing throttle mode. The supported IPMI commands are documented in the following sections. Standard IPMI details are not documented in this document. Refer to the *IPMI Specification, v2.0.* For example, the Get SDR command requires additional bytes to complete the command packet and these bytes are defined in the *IPMI Specification, v2.0*.

6.5.3.1 Miscellaneous Commands

Table 6-2. Miscellaneous Command Details

NetFn	Command	Name
Chassis (0x00)	0x02	Chassis Control
App (0x06)	0x01	Get Device ID
App (0x06)	0x08	Get Device GUID (UUID)

6.5.3.2 FRU Related Commands

Table 6-3. FRU Related Command Details

NetFn	Command	Name
Storage (0x0a)	0x10	Get FRU Inventory Area Info
Storage (0x0a)	0x11	Read FRU Data

6.5.3.3 SDR Related Commands

Table 6-4. SDR Related Command Details

NetFn	Command	Name
Storage (0x0a)	0x20	Get SDR Repository Info
Storage (0x0a)	0x21	Get SDR Repository Allocation Info
Storage (0x0a)	0x23	Get SDR

Note: The SDR can be read in "chunks", suggested size is 16 bytes, or the entire SDR can be read by passing 'FF' as the number of bytes to read.

6.5.3.4 SEL Related Commands

Table 6-5. SEL Related Command Details

NetFn	Command	Name
Storage (0x0a)	0x40	Get SEL Info
Storage (0x0a)	0x41	Get SEL Allocation Info
Storage (0x0a)	0x43	Get SEL Entry
Storage (0x0a)	0x47	Clear SEL
Storage (0x0a)	0x48	Get SEL Time
Storage (0x0a)	0x49	Set SEL Time

6.5.3.5 Sensor Related Commands

Table 6-6. Sensor Related Command Details

NetFn	Command	Name
Sensor (0x04)	0x2b	Get Sensor Event Status
Sensor (0x04)	0x2d	Get Sensor Reading

6.5.3.6 General Commands

Table 6-7. General Command Details

NetFn	Command	Name
Intel (0x2e)	0x42	CPU Package Config Read
Intel (0x2e)	0x43	CPU Package Config Write
Intel General App (0x30)	0x15	Set SM Signal

6.5.3.6.1 CPU Package Configuration Read

The CPU Package Config Read command reads power control data.

Table 6-8. CPU Package Config Read Request Format

Byte #	Value	Description
Command	0x42	CPU Package Config Read
NetFn	0x2e	NETFN_INTEL
0-2		Manufacturer ID (LSB format): 0x57, 0x01, 0x00
3	0x00	CPU Number
4	0x??	PCS Index • 3 - Accumulated Energy Status • 11 - Socket Power Throttle Duration • 26 - Package Power Throttle Threshold Value 1 (PL1) • 27 - Package Power Throttle Threshold Value 2 (PL0) • 28 - Package Power SKU A • 29 - Package Power SKU B • 30 - Package Power SKU Unit • All other values reserved
5	0x00	Parameter LSB
6	0x00	Parameter MSB
7	0x??	Number of Bytes to Read

Table 6-9. CPU Package Config Read Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal • 0xcc - Invalid field • 0xa1 - Wrong CPU Number • 0xa7 - Wrong Read Length • 0xab - Wrong Command Code • 0xff - Unspecified Error
1-3		Manufacturer ID (LSB format): 0x57, 0x01, 0x00
4[-7]	0x??	Data bytes read, up to 4 bytes

6.5.3.6.2 CPU Package Configuration Write

The CPU Package Config Write command allows the setting of power control data.

Table 6-10. CPU Package Config Write Request Format

Byte #	Value	Description
Command	0x43	CPU Package Config Write
NetFn	0x2e	NETFN_INTEL
0-2		Manufacturer ID (LSB format): 0x57, 0x01, 0x00
3	0x00	CPU Number
4	0x??	PCS Index • 26 - Package Power Throttle Threshold Value 1 (PL1) • 27 - Package Power Throttle Threshold Value 2 (PL0) • All other values reserved
5	0x00	Parameter LSB
6	0x00	Parameter MSB
7	0x??	Number of Bytes to Write
8[-11]	0x??	Data bytes to write

Table 6-11. CPU Package Config Write Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal • 0xc7 - Request Length Invalid • 0xcc - Invalid Field • 0xa1 - Wrong CPU Number • 0xa6 - Wrong Write Length • 0xab - Wrong Command Code • 0xff - Unspecified Error
1-3		Manufacturer ID (LSB format): 0x57, 0x01, 0x00

6.5.3.6.3 Set SM Signal

The Set SM Signal command gives you control of firmware signals. The primary use of this command is to set the status LED into identify mode. In identify mode, the status LED flashes on for a short period twice every 2 seconds. This allows an administrator to locate the coprocessor in a system that has multiple coprocessors.

Table 6-12. Set SM Signal Request Format (Sheet 1 of 2)

Byte #	Value	Description
Command	0x15	Set SM Signal
NetFn	0x30	NETFN_INTEL_GENERAL_APP
0	0x??	Signal • 1 - Identify • All other values reserved

Table 6-12. Set SM Signal Request Format (Sheet 2 of 2)

Byte #	Value	Description
1	0x00	Instance
2	0x??	Action If Signal is 1 • 1 - Assert: Start the identify blink code • 2 - Revert: Return to normal operation • All other values reserved
[3]	0x00	Value (optional)

Table 6-13. Set SM Signal Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal • 0xc7 - Request Length Invalid • 0xc9 - Parameter Out of Range • 0xcc - Invalid Field

6.5.3.7 OEM Commands

Table 6-14. OEM Command Details

NetFn	Command	Name
OEM (0x3e)	0x00	OEM Set Fan PWM Adder
OEM (0x3e)	0x04	OEM Get POST Register
OEM (0x3e)	0x05	OEM Assert Forced Throttle
OEM (0x3e)	0x06	OEM Enable External Throttle

6.5.3.7.1 OEM Set Fan PWM Adder

The Set Fan PWM Adder command allows a PWM percentage to be added to the final fan cooling algorithm for additional cooling based on chassis requirements.

Table 6-15. Set Fan PWM Adder Command Request Format

Byte #	Value	Description
Command	0x00	OEM Set Fan PWM Adder
NetFn	0x3e	NETFN_OEM
0	0x??	PWM percent to add to standard cooling • 0x00 - 0x64 • All other values are reserved.

Table 6-16. Set Fan PWM Adder Command Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal • 0xc9 - Parameter out of range

6.5.3.7.2 OEM Get POST Register

The Get POST Register command allows the BMC to obtain the last POST code written to the SMC by the coprocessor. The SMC does not modify this value in any way.

Table 6-17. Get POST Register Request Format

Byte #	Value	Description
Command	0x04	OEM Get POST Register
NetFn	0x3e	NETFN_OEM

Table 6-18. Get POST Register Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal
1-4	0x??	32 bit POST code in little endian format

6.5.3.7.3 OEM Assert Forced Throttle

The Assert Forced Throttle command allows the BMC to cause the SMC to assert the PROCHOT_N pin to the coprocessor.

Table 6-19. Assert Forced Throttle Request Format

Byte #	Value	Description
Command	0x05	OEM Assert Forced Throttle
NetFn	0x3e	NETFN_OEM
0	0x??	• 0 - Deassert forced throttle • 1 - Assert forced throttle • All other values are reserved

Table 6-20. Assert Forced Throttle Response Format

Byte #	Value	Description
0	0x??	Compcode • 0x00 - Normal

6.5.3.7.4 OEM Enable External Throttle

The Enable External Throttle command causes the SMC to enable a pin on the PCIe connector (pin B12 or B30) allowing the baseboard (host) BMC to directly assert the PROCHOT_N signal. The requirements to enable this pin on the baseboard are described in Section 4.1.1.

The signal to assert emergency throttling via pin B12 or B30 is active low on the baseboard and is driven by the BMC. However, the pin must first be enabled by the SMC. This can be accomplished by sending the Enable External Throttle command as described in this section. The pin needs to be enabled each time a reset or power cycle event occurs. Its state is not persistent across these events.

When the baseboard asserts PROCHOT_N (drives active low signal), the coprocessor OS immediately drops the frequency to lowest rated value (Pn) within 100 µs of asserting PROCHOT_N. If PROCHOT_N is deasserted in less than 100 ms, the coprocessor frequency is restored to the original operational value (either P1 or turbo). If baseboard continues to assert PROCHOT_N for more than 100 ms, the coprocessor OS responds by reducing the voltage ID (VID) settings to match the lowest frequency, leading to further power savings. Upon subsequent deassertion of PROCHOT_N, the VID settings are first restored to support operational frequency, followed by the coprocessor frequency itself.

If a baseboard does not support the B12 or B30 capability, the external throttle signals can be disabled using this command. The coprocessor can still be throttled using the SMC by sending the Assert Forced Throttle Command referenced above.

Table 6-21. Enable External Throttle Request Format

Byte	Value	Description
Command	0x06	OEM Enable External Throttle
NetFn	0x3e	NETFN_OEM
0	0x??	• 0x00 - Disable external throttle signal • 0x01 - Enable external throttle signal and select B12 • 0x11 - Enable external throttle signal and select B30 • All other values are reserved

Table 6-22. Enable External Throttle Response Format

Byte	Value	Description
0	0x??	Compcode • 0x00 - Normal • 0xc0 - Busy

6.5.3.8 Other IPMI Related Information

The SMC SEL is a circular log supporting a minimum of 64 log entries. It is resilient to corruption, retaining information across an unexpected power loss.

The sensor names in the IPMI SDR are static and do not change from release to release. The IPMI sensor numbers are not static and may change between releases; hence the sensor number should be discovered during the normal sensor discovery process because additional sensors may be added in the future.

During the normal sensor discovery process, reading the SDR returns the sensors available on the coprocessor. There is a sensor name and sensor number associated with each sensor. Once the sensor number is determined by comparing the sensor name that is desired to be read, the sensor number may be used by the management firmware for reading a particular sensor. It is important that the firmware does not hard-code the sensor number as it may change in future releases. Instead, Intel recommends that you discover the sensor number using the sensor name to ensure the correct sensor is read and returns valid data with future releases of the SMC firmware. Table 6-23 is a list of the current sensor names.

Table 6-23. Table of Sensors (Sheet 1 of 2)

Signal	Signal Description
STATUS	
status	Status Bits
TEMPERATURE	
vr_temp	Temperature sensor in VR area of board
airflow_temp	Temperature sensor in airflow out (exhaust) area of board
cpu_temp	Temperature sensor in CPU (processor) area of board
ntb_temp	Temperature sensor in NTB area of board
vccp_temp	Temperature reported from the VCCP VR
vccclr_temp	Temperature reported from the VCCCLR VR
vccmp_temp	Temperature reported from the VCCMP VR
proc_temp	Temperature reported by the CPU die
exhst_temp	Highest of the discrete temperature sensors on the board
inlet_temp	Lowest of the discrete temperature sensors on the board
VOLTAGE	
vccp_volt	Voltage reported by the VCCP VR
vccu_volt	Voltage reported by the VCCU VR
vccclr_volt	Voltage reported by the VCCCLR VR
vccmlb_volt	Voltage reported by the VCCMLB VR
vccmp_volt	Voltage reported by the VCCMP VR
ntb1_volt	Voltage reported by the NTB VR
vccpio_volt	Voltage obtained from the VCCPIO rail
vccsfr_volt	Voltage obtained from the VCCSFR rail
pch_volt	Voltage obtained from the PCH rail
vccmfuse_volt	Voltage obtained from the VCCMFUSE rail
ntb2_volt	Voltage obtained from the NTB rail
vpp_volt	Voltage obtained from the VPP rail
POWER	
power_pcie	Power measured at the PCIe* edge fingers input
power_2x3	Power measured at the 2x3 auxiliary power connector input
power_2x4	Power measured at the 2x4 auxiliary power connector input
avg_power0	Average power consumed over a time window of 50 ms
instpwr	Instantaneous power consumption reading
instpwrmax	Maximum instantaneous power consumption observed
persistpwr	Maximum power observed, obtained from non-volatile storage
power_vccp	Power output reported by the VCCP VR
power_vccu	Power output reported by the VCCU VR
power_vccclr	Power output reported by the VCCCLR VR
power_vccmlb	Power output reported by the VCCMLB VR
power_vccmp	Power output reported by the VCCMP VR
power_ntb1	Power output reported by the NTB VR

Table 6-23. Table of Sensors (Sheet 2 of 2)

Signal	Signal Description
FAN	
fan_pwm	Fan PWM driven by SMC software (N/A for passive SKUs)
fan_tach	Fan tachometer value (N/A for passive SKUs)

6.5.3.9 SMC IPMI Discrete Sensors

The SMC's IPMI discrete sensors are defined here because the meaning of each discrete bit cannot be easily derived from the SDR definition.

6.5.3.9.1 Sensor Status

The status sensor reports the state of several critical signals on the coprocessor such as Thermtrip, VR hot, and PCI Express Reset. The sensor is not mirrored as a register on the in-band register interface.

6.6 SMC LED_ERROR and Fan PWM

Table 6-24. Status Sensor Report Format

Bits	Name	Description
15:8	RESERVED	
7	VCCMP_VR_HOT	VCCMP VR Hot signal asserted. Fans boosted and PROCHOT asserted.
6	P2E_RST	PCIe reset asserted. Fans boosted.
5	RESERVED	
4	VCCLR_VR_HOT	VCCCLR VR Hot signal asserted. Fans boosted and PROCHOT asserted.
3	VCCP_VR_HOT	VCCP VR Hot signal asserted. Fans boosted and PROCHOT asserted.
2	RESERVED	
1	RESERVED	
0	THERMTRIP	CPU Thermtrip asserted. Fans boosted and VR output disabled. This state is latched until power-off.

The SMC firmware drives the LED_ERROR pin as follows.

Table 6-25. LED Indicators

LED Color	Blink Frequency	Condition
Green	Activity 0.5 Hz Blink	• In boot loader mode
Green	Activity 2H z Blink	• Firmware update in progress
Green	Activity 8 Hz Blink	• Operational code executing
Blue	Identify Blink	• 2 short blinks every 2 seconds • Initiated by SetSMSignal command.

§

Appendix A Platform Reset Considerations

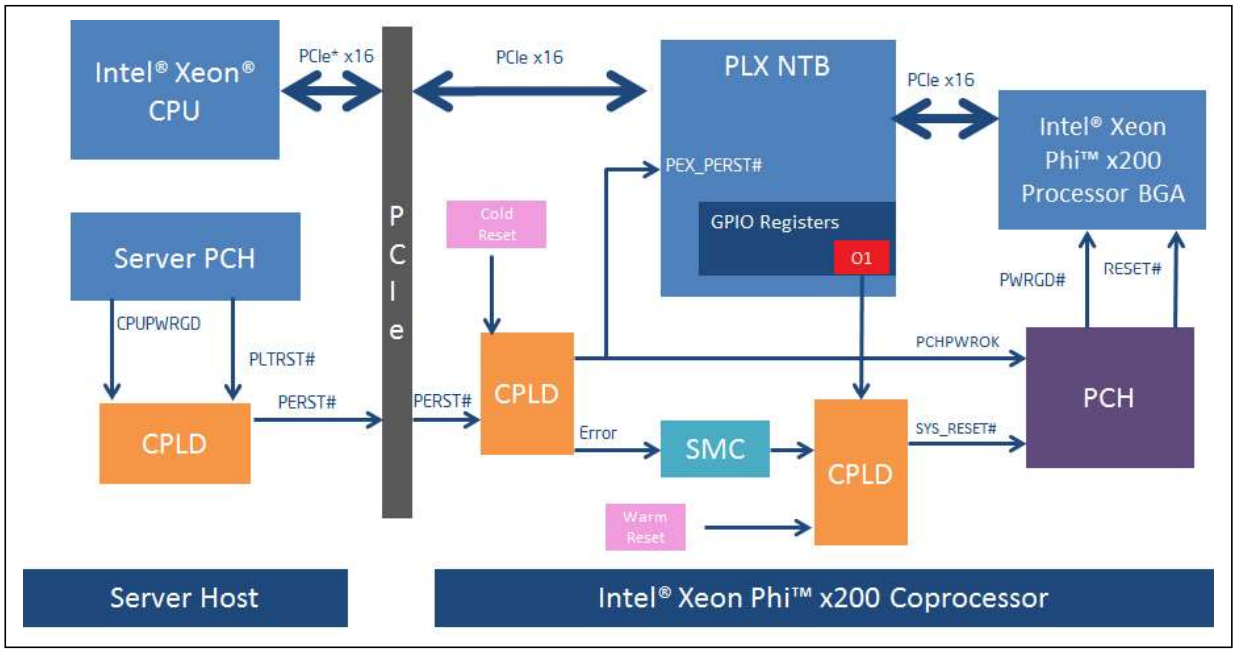
Upon detecting an error, the coprocessor logs error details. These error logs are cleared on cold reset or PERST# assertion. Most host platforms drive PERST# with the general platform reset signal PLTRST#. PLTRST# is asserted on both a host warm and cold reset.

When the host goes through a warm reset, the coprocessor goes through a cold reset. To avoid losing the error logs during a host warm reset, Intel recommends connecting the PERST# signal to the host CPUPWRGD signal. This causes the coprocessor to get a cold reset only when the host gets a cold reset. See Figure A-1 for an example of this topology.

Host platforms can use a jumper or Programmable Logic Device (PLD) to select between PLTRST# or CPUPWRGD to drive the co-processor's PERST# signal. To issue a warm reset to the coprocessor, the host platform must assert and de-assert the NTB GPIO1 pin.

1. When the driver sets the PLX8733 GPIO1 bit to 1, the PLD asserts SYS_RESET# to the PCH.
2. A SYS_RESET# to the PCH activates a warm reset sequence for the processor and the PCH.
3. The host driver must clear the GPIO1 to 0 after 30 – 50 ms after assertion.

Figure A-1. Topology Example of PERST#



§