Intel® Core™ Processor (Series 3)

Formerly known as Wildcat Lake, Datasheet, Volume 1 of 2

Rev. 001

April 2026

Doc. No.: 913965, Rev.: 001

Contents

Figures

Tables

Revision History

Document Number	Revision Number	Description	Revision Date
913965	001	Initial Release	April 2026

1.0 Introduction

The Intel® Core™ Processor (Series 3), formerly known as Wildcat Lake processor is a 64-bit, multi-core processor built on Intel 18A process, N6 and N3E technologies.

The Intel® Core™ Processor (Series 3) is offered in a single package platform that includes the Compute tile, and the PCD tile.

The following table describes the offered Intel® Core™ Processor (Series 3):

Table 1. Intel® Core™ Processor (Series 3) Form Factors

Form Factors[1]	Package	Processor Base Power [2, 3]	P-cores	E-cores	LP E-cores	Graphics Configuration Xe-cores[4]	Platform Type
6C	BGA1516	15W	up to 2	0	4	up to 2	1-Chip

Notes:
1. Form Factors offering may change.
2. For additional Processor Base Power Configurations, refer to Processor Base Power Thermal and Power Specifications on page 122. For adjustment to the Processor Base Power it is required to preserve base frequency associated with the sustained long-term thermal capability.
3. Processor Base Power workload does not reflect I/O connectivity cases such as Thunderbolt.

1.1 Processor Volatility Statement

Intel® Core™ Series 3 processor families do not retain any end-user data when powered down and/or when the processor is physically removed.

NOTE

Powered down refers to the state in which all processor power rails are off.

1.2 Package Dimensions

1.2.1 Intel® Core™ Processor (Series 3) Package Details

The Intel® Core™ Processor (Series 3) is available in the following package:

- BGA 1516
- A 35 x 25 mm
- Substrate Z-height = 473±60 um
- Total Package Z-height (Bottom of BGA to top) = 1.055 ± 0.09 mm

1.3 Supported Technologies

- PCI Express* (PCIe*)
- Flexible High Speed I/O
- Xe3 Graphics Core Based Processor Graphics
- Display
  - VESA Certified DisplayPort (v2.1, HBR3, DSC)
  - Embedded DisplayPort* 1.4 (eDP* 1.4)
  - Embedded DisplayPort* 1.5 (eDP* 1.5)
  - High-Definition Multimedia Interface* 2.1 (HDMI* 2.1)
- Intel® Neural Processing Unit (Intel® NPU)
- USB Type-C* Sub System
  - Intel® Thunderbolt™
  - USB 3.2 Gen 2x1 (10 Gb/s) eXtensible Host Controller (xHCI)
  - USB 3.2 Gen 2x2 (20 Gb/s) eXtensible Host Controller (xHCI)
  - VESA Certified DisplayPort (v2.1, HBR3, DSC)
- Memory
  - DDR5
  - LPDDR5x
  - In Band Error Code Correction (IBECC)
  - Intel® In-Memory Analytics Accelerator (Intel® IAA)
- Platform Environmental Control Interface (PECI) over eSPI
- Universal Serial Bus (USB)
- Universal Flash Storage (UFS)
- Intel® Volume Management Device (Intel® VMD)
- Touch Host Controller (THC)
- Intel® Serial I/O I^2C
- Intel® Serial IO Improved Inter-Integrated Circuit (I^3C) Controllers
- Integrated Sensor Hub (ISH)
- Integrated Connectivity (CNVi)
- Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers
- Intel® Serial IO Generic SPI (GSPI) Controllers
- Serial Peripheral Interface (SPI)
- Enhanced Serial Peripheral Interface (eSPI)
- Intel® High Definition Audio (Intel® HD Audio)
- Intel® Smart Sound Technology (Intel® SST)
- System Management Bus (SMBus)
- Gigabit Ethernet (GbE) controller
- Integrated Clock Controller (ICC)/Integrated Reference Clock PLL

- Real Time Clock Controller (RTCC)
- General Purpose Input Output (GPIO)
- Controller Link
- System Management Interface and SMLink
- Virtualization Technologies
  - Intel® Virtualization Technology (Intel® VT-x)
  - Hypervisor-Managed Linear Address Translation (HLAT)
  - Intel® Virtualization Technology for Directed I/O (Intel® VT-d)
  - Intel® APIC Virtualization Technology (Intel® APICv)
- Security Technologies
  - Intel® Trusted Execution Technology (Intel® TXT)
  - Intel® Converged Boot Guard and Intel® Trusted Execution Technology (Intel® CBnT)
  - Intel® Secure Key
  - Execute Disable Bit
  - Intel® Supervisor Mode Execution Protection (Intel® SMEP)
  - Intel® Supervisor Mode Access Protection (Intel® SMAP)
  - User Mode Instruction Prevention (UMIP)
  - Read Processor ID (RDPID)
  - Intel® Total Memory Encryption (Intel® TME)
  - Intel® Multi-Key Total Memory Encryption (Intel® MK-TME)
  - Intel® Total Storage Encryption (Intel® TSE)
  - Intel® Control-flow Enforcement Technology (Intel® CET)
  - Linear Address Space Separation (LASS)
  - Intel® System Resources Defense and Intel® System Security Report
  - Intel® Debug Protection
  - Intel® BIOS Guard
  - Intel® Boot Guard
  - Intel® Platform Trust Technology (Intel® PTT)
  - Intel® Platform Firmware Resiliency (Intel® PFR)
- Security Technologies - Crypto Acceleration Instructions
  - Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)
  - Perform Carry-Less Multiplication Quad Word Instruction (PCLMULQDQ)
  - Intel® Secure Hash Algorithm - 512 (Intel® SHA - 512)
  - Intel® Secure Hash Algorithm Extensions (Intel® SHA Extensions)
- Security Technologies - Security Firmware Engines
  - Intel® Converged Security and Management Engine (Intel® CSME)
  - Intel® Silicon Security Engine

  - — Intel® Graphics System Controller (Intel® GSC)
  - — Intel® Active Management Technology (Intel® AMT)
  - — Intel® Partner Security Engine
- Testability and Monitoring
  - — JTAG Boundary Scan
  - — Intel® Software Toolkit
  - — Platform Crashlog
  - — Platform Monitoring Technology (PMT)
  - — Intel® Processor Trace
  - — Intel® Trace Hub (Intel® TH)
  - — Direct Connect Interface (DCI) for debug
  - — Debug Island
  - — Early Boot Debug
- Power Management Technologies
  - — Advanced Configuration and Power Interface (ACPI) Power Management Logic Support
  - — Intel® Smart Cache Technology
  - — Power and Efficient Cores Level 1 and Level 2 Caches
  - — Cache Line Write Back (CLWB)
  - — Intel® Thread Director
  - — Intel® Hybrid Technology
  - — Intel® Turbo Boost Technology 2.0
  - — Intel® Turbo Boost Max Technology 3.0
  - — Intel® Adaptive Boost Technology
  - — Intel SpeedStep® Technology
  - — Intel® System Agent Enhanced SpeedStep Technology (Intel® SAGV)
  - — Intel® Speed Shift Technology
  - — Intel® Advanced Vector Extensions 2 (Intel® AVX2)
  - — Intel® AVX2 Vector Neural Network Instructions (Intel® AVX2 VNNI)
  - — Intel® Advanced Programmable Interrupt Controller (APIC)
  - — Intel® 64 Architecture x2APIC
  - — Intel® Dynamic Tuning Technology (Intel® DTT)
- Power Delivery Technologies
  - — Thermally Equal Turbo-boost Algorithm (ThETA)
  - — Digital Linear Voltage Regulator (DLVR)
  - — Current Excursion Protection (CEP)
  - — Reactive PL4 with PL4 Boost
  - — Thermally Equal Turbo-Boost Algorithm (ThETA)

NOTE

The availability of the features above may vary between different processor SKUs.

Deprecated Technology

- Intel® Volume Management Device (Intel® VMD)

1.3.1 API Support (Windows)

- DirectML, Direct3D 12.2, Direct3D 12.1, Direct3D 12, Direct3D 11.4, Direct3D 11.3, Direct3D 11.2, Direct3D 11.1, Direct3D 10.1, Direct3D 10, Direct3D 9.0L, Direct3D 9.0C, Direct2D
- WinML, MEP
- OpenGL* 4.6
- Vulkan 1.3
- Open CL* 3.0, Open CL* 2.1, Open CL 2.0, Open CL 1.2

DirectX* extensions:

- PixelSync, Instant Access, Conservative Rasterization, Render Target Reads, Floating-point De-norms, Shared a Virtual memory, Floating Point atomics, MSAA sample-indexing, Fast Sampling (Coarse LOD), Quilted Textures, GPU Enqueue Kernels, GPU Signals processing unit. Other enhancements include color compression.

Xe architecture delivers hardware acceleration of Direct X* 12.2 Render pipeline comprising of the following stages: Vertex Fetch, Vertex Shader, Hull Shader, Tessellation, Domain Shader, Geometry Shader, Rasterizer, Pixel Shader, Pixel Output, Mesh shading, Variable rate shading, Sampler feedback.

1.4 Power Management Support

1.4.1 Processor Core Power Management

Full support of ACPI C-states as implemented by the following processor C-states:

- C0, C1, C1E, C6, C10

1.4.2 Memory Controller Power Management

- Disabling Unused System Memory Outputs
- DRAM Power Management and Initialization
- Clock Enable (CKE)
- Conditional Self-Refresh
- Dynamic Power Down
- DRAM I/O Power Management
- DDR Electrical Power Gating (EPG)
- Power Training

1.4.3 Processor Graphics Power Management

Graphics Memory Power Savings Technologies

- Intel® Rapid Memory Power Management (Intel® RMPM)

Graphics Display Power Savings Technologies

- Intel® Seamless Display Refresh Rate Switching (Intel® DRRS) with eDP* port
- Intel® Display Power Saving Technology (Intel® DPST) 8.0
- Intel® OLED Power Saving Technology (Intel® OPST) 1.1
- Panel Self-Refresh 2 (PSR 2)
- Low-Power Single Pipe (LPSP)
- Low-Power Dual Pipe (LPDP)
- Intel® Smart 2D Display Technology (Intel® S2DDT)
- Intel® Low Refresh Rate (LRR)

Graphics Core Power Savings Technologies

- Intel® Graphics Dynamic Frequency
- Intel® Graphics Render Standby Technology (Intel® GRST)
- Intel® Capped Frames Per Second (Intel® CFPS)

1.5 Thermal Management Support

- Intel® Adaptive Thermal Monitor
- Digital Thermal Sensor
- THERMTRIP# and FORCEPR# support
- Critical Temperature Detection
- Software Controlled Clock Modulation (On-Demand Mode)
- Memory Thermal Throttling
- Render Thermal Throttling
- Fan Speed Control with DTS

1.6 Ballout Information

For information on the Intel® Core™ Processors (Series 3) processor ball information, refer to the PDF download. Click on the navigation pane and refer to the spreadsheet 913965-001_Ballout.xlsx.

1.7 Operating Systems Support

Windows* 11 OS	Chrome* OS	Linux* OS
Yes	Yes	Yes[1]
Note: 1. Partial ingredient level support.		

NOTE

Refer to OS Vendor site for more information regarding latest OS revision support.

1.8 Terminology and Special Marks

Terminology Usage

This document uses the term **Processor** to indicate the **Compute Tile + PCD Tile** . Individually, **Compute Tile** , **PCD Tile** are used.

Table 2. Terminology

Term	Description
4K	Ultra High Definition (UHD)
AES	Advanced Encryption Standard
AGC	Adaptive Gain Control
API	Application Programming Interface
AVC	Advanced Video Coding
BLT	Block Level Transfer
BPP	Bits per Pixel
CDR	Clock and Data Recovery
CTLE	Continuous Time Linear Equalizer
D0ix-states	USB controller power states ranging from D0i0 to D0i3, where D0i0 is fully powered on and D0i3 is primarily powered off. Controlled by SW.
DDC	Digital Display Channel
DDI	Digital Display Interface for DisplayPort or HDMI/DVI
DFE	Decision Feedback Equalizer
DMA	Direct Memory Access
DPPM	Dynamic Power Performance Management
DP*	DisplayPort*
DSC	Display Stream Compression
DTS	Digital Thermal Sensor
EU	Execution Unit in the Graphics Processor
GSA	Graphics in System Agent
HDCP	High-Bandwidth Digital Content Protection
IMC	Integrated Memory Controller
Intel® 64 Technology	64-bit memory extensions to the IA-32 architecture
Intel® DPST	Intel® Display Power Saving Technology
Intel® PTT	Intel® Platform Trust Technology

continued...

Term	Description
Intel® TXT	Intel® Trusted Execution Technology
Intel® VT	Intel® Virtualization Technology. Processor Virtualization, 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, under system software (Virtual Machine Manager or OS) control, for enabling I/O device Virtualization. Intel® VT-d also brings robust security by providing protection from errant DMAs by using DMA remapping, a key feature of Intel® VT-d.
Intel® TH	Intel® Trace Hub
IOV	I/O Virtualization
LFM	Low Frequency Mode. corresponding to the Enhanced Intel SpeedStep® Technology's lowest voltage/frequency pair. It can be read at MSR CEh [47:40]. .
LLC	Last Level Cache
LPDDR5x	Fifth generation Low Power Double Data Rate SDRAM memory technology, x- high frequency.
LPSP	Low-Power Single Pipe
LSF	Lowest Supported Frequency.This frequency is the lowest frequency where manufacturing confirms logical functionality under the set of operating conditions.
LTR	The Latency Tolerance Reporting (LTR) mechanism enables Endpoints to report their service latency requirements for Memory Reads and Writes to the Root Complex, so that power management policies for central platform resources (such as main memory, RC internal interconnects, and snoop resources) can be implemented to consider Endpoint service requirements.
MFM	Minimum Frequency Mode. MFM is the minimum ratio supported by the processor and can be read from MSR CEh [55:48].
MLC	Mid-Level Cache
MoP	Memory on Package
MPEG	Motion Picture Expert Group, international standard body JTC1/SC29/WG11 under ISO/IEC that has defined audio and video compression standards such as MPEG-1, MPEG-2, and MPEG-4, etc.
NCTF	Non-Critical to Function. NCTF locations are typically redundant ground or non-critical reserved balls/lands, so the loss of the solder joint continuity at end of life conditions will not affect the overall product functionality.
PEG	PCI Express* Graphics
PL1, PL2, PL3, PL4	Power Limit 1, Power Limit 2, Power Limit 3, Power Limit 4
PMIC	Power Management Integrated Circuit
Processor	The 64-bit multi-core component (package)
Processor Core	The term "processor core" refers to the Si tile itself, which can contain multiple execution cores. Each execution core has an instruction cache, data cache, and 256-KB L2 cache. All execution cores share the LLC. P core - performance cores, and LP E-core- efficiency cores.
Processor Graphics	Intel® Processor Graphics
PSR	Panel Self-Refresh
PSx	Voltage regulator power states (PS0, PS1, PS2)
SCI	System Control Interrupt. SCI is used in the ACPI protocol.

continued...

Term	Description
SDP	Scenario Design Power
SVID	Serial Voltage IDentification Code HW and SW protocol used by CPU to dynamically control a variable Voltage regulator
SHA	Secure Hash Algorithm
SSC	Spread Spectrum Clock
TAC	Thermal Averaging Constant
T&L	Thin and Light
TBT	Thunderbolt™ Interface
TCC	Thermal Control Circuit
TTV Processor Base Power	Thermal Test Vehicle Processor Base Power
VCC_PCORE, VCC_ECORE, VCCL2	Processor Core Power Supply
V_{CCGT}	Processor Graphics Power Supply
V_{CCSA}	System Agent Power Supply
VLD	Variable Length Decoding
VPID	Virtual Processor ID
V_{SS}	Processor Ground

Table 3. Special Marks

Mark	Definition
[]	Brackets ([]) sometimes follow a ball, pin, registers or a bit name. These brackets enclose a range of numbers, for example, TCP[2:0]_TXRX_P[1:0] may refer to four USB-C* pins or EAX[7:0] may indicate a range that is 8 bits length.
_N / #	A suffix of _N or # indicates an active low signal. For example, CATERR# _N does not refer to a differential pair of signals such as CLK_P, CLK_N
h	Hexadecimal numbers are identified with an h in the number. All numbers are decimal (base 10) unless otherwise specified. Non-obvious binary numbers have the 'b' enclosed at the end of the number. For example, 0101b

1.9 Flexible High Speed IO

Flexible Input/Output (I/O) is a technology that allows the High Speed I/O (HSIO) lanes to be configured for connection to a Gigabit Ethernet (GbE) Controller, a PCIe* Controller, or an Extensible Host Controller Interface (xHCI) USB 3.2 Controller. Flexible I/O enables customers to optimize the allocation of the HSIO interfaces to better meet the I/O needs of their system.

NOTE

Some Flexible I/O multiplexing capabilities are not available on all SKUs.

Table 4. Acronyms

Acronyms	Description
USB	Universal Serial Bus
PCIe*	PCI Express* (Peripheral Component Interconnect Express*)
GbE	Gigabit Ethernet
HSIO	High Speed Input/Output

1.9.1 Intel® Core™ Series 3 Flexible HSIO Lanes

Figure 1. Flexible HSIO Lane Details

Intel® Core™ Series 3	Platform Controller Die (PCD)								Max Device Suppor	
Flex I/O Lane	1	2	3	4	5	6	7	8		
USB 3.2 Lanes	1	2							2	
PCIe 4.0 Lanes			A1	A2	A3	A4	C1	C2	6	5
GbE Lanes								X	0	1

The 8 Flexible HSIO Lanes [10:1] support the following:

1. Up to 6 PCIe* Lanes
   - A maximum of six PCIe* Root Ports (or devices) can be enabled when GbE Port is disabled.
   - A maximum of five PCIe* Root Ports (or devices) can be enabled when GbE Port is enabled .
   - PCIe* Lanes A1-A4 (PCIe* Controller A) and C1-C2 (PCIe* Controller C) must be individually configured.
2. Up to two USB 3.2 Gen 1x1/2x1 Lanes
   - A maximum of two USB 3.2 Gen 1x1/2x1 Ports (or devices) can be enabled
   - USB 3.2 Gen 1x1 = First Generation with One 5 GT/s Data Lane
   - USB 3.2 Gen 2x1 = Second Generation with One 10 GT/s Data Lane
3. Up to one GbE Lane
   - A maximum of one GbE Port can be enabled

2.0 Processor and Device IDs

2.1 CPUID

Table 5. CPUID Format

SKU	CPUID	Reserved [31:28]	Extended Family [27:20]	Extended Model [19:16]	Reserved [15:14]	Processor Type [13:12]	Family Code [11:8]	Model Number [7:4]	Stepping ID [3:0]
6C A1 Step	D0651 h	0000b	0000000b	1101b	00b	00b	0110b	0101b	0001b

- The Extended Family, Bits [27:20] are used in conjunction with the Family Code, specified in Bits[11:8], to indicate whether the processor belongs to Intel® Core™ processor family.
- The Extended Model, Bits [19:16] in conjunction with the Model Number, specified in Bits [7:4], are used to identify the model of the processor within the processor's family.
- The Family Code corresponds to Bits [11:8] of the EDX register after RESET, Bits [11:8] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the generation field of the Device ID register accessible through Boundary Scan.
- The Model Number corresponds to Bits [7:4] of the EDX register after RESET, Bits [7:4] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the model field of the Device ID register accessible through Boundary Scan.
- The Stepping ID in Bits [3:0] indicates the revision number of that model.

Cache and TLB descriptor parameters are provided in the EAX, EBX, ECX and EDX registers after the CPUID instruction is executed with a 2 in the EAX register.

2.2 PCI Configuration Header

Every PCI-compatible function has a standard PCI configuration header, as shown in the table below. This includes mandatory registers (Bold) to determine which driver to load for the device. Some of these registers define ID values for the PCI function, which are described in this chapter.

Table 6. PCI Configuration Header

Byte3	Byte2	Byte1	Byte0	Address
Device ID		**Vendor ID (8086h)**		00h
Status		Command		04h
Class Code			**Revision ID**	08h
BIST	Header Type	Latency Timer	Cache Line Size	0Ch

continued...

Byte3	Byte2	Byte1	Byte0	Address
Base Address Register0 (BAR0)				10h
Base Address Register1 (BAR1)				14h
Base Address Register2 (BAR2)				18h
Base Address Register3 (BAR3)				1Ch
Base Address Register4 (BAR4)				20h
Base Address Register5 (BAR5)				24h
Card-bus CIS Pointer				28h
Subsystem ID		**Subsystem Vendor ID**		2Ch
Expansion ROM Base Address				30h
Reserved			Capabilities Pointer	34h
Reserved				38h
Maximum Latency	Minimum Grant	Interrupt Pin	Interrupt Line	3Ch

2.3 Device IDs

This section specifies the device IDs of the processor.

Table 7. Host Device ID (DID0) and Processor Graphics Device ID (DID2)

Processor Line	Package	Compute Tile P-Cores	Compute Tile LP E-Cores	Graphics Configuration X^e-cores	Host Device ID (DID0)	Processor Graphics Device ID (DID2)
6C	1516	2	4	2	FD00h	FD80h

Table 8. Other Device ID

Device	Bus / Device / Function	6C Processor (DID)
Dynamic Tuning Technology (DTT)	0 / 4 / 0	FD1Dh
PCI Express Root Port #5	0 / 6/ 0	4D61h
PCI Express Root Port #6	0 / 6 / 1	4D5Ch
USB Type-C Subsystem PCIe Root Port #21	0 / 7 / 0	4D4Eh
USB Type-C Subsystem PCIe Root Port #22	0 / 7 / 1	4D4Fh
Intel® Platform Monitoring Technology (Intel® PMT) Intel® Crash Log Technology and Telemetry	0 / 10 / 0	FD7Dh
NPU	0/ 11 / 0	FD3Eh
IAA (IAX)	0/ 12 / 0	FD2Dh
Type-C Subsystem xHCI	0 / 13 / 0	4D31h
Thunderbolt™ DMA0	0 / 13 / 2	4D33h
THC #0 (Touch Host Controller) ID1	0 / 16 / 0	4D48h

continued...

Device	Bus / Device / Function	6C Processor (DID)
THC #0 (Touch Host Controller) ID2	0 / 16 / 0	4D49h
THC #1 (Touch Host Controller) ID1	0 / 16 / 1	4D4Ah
THC #1 (Touch Host Controller) ID2	0 / 16 / 1	4D4Bh
I3C Controller #1	0 / 17 / 0	4D7Ch
I3C Controller #2	0 / 17 / 1	4D6Fh
Integrated Sensor Hub (ISH)	0 / 18 / 0	4D45h
P2SB (16 bit)	0 / 18 / 1	4D4Ch
IEH #1	0 / 18 / 3	4D53h
GSPI #2	0 / 18 / 6	4D46h
Intel® CSME: HECI #1	0 / 19 / 0	4D62h
Intel® CSME: HECI #2	0 / 19 / 1	4D63h
Intel® CSME: HECI #3	0 / 19 / 2	4D64h
Standalone xHCI Controller	0 / 20 / 0	4D7Dh
Standalone USB Device Controller	0 / 20 / 1	4D7Eh
Shared SRAM	0 / 20 / 2	4D7Fh
CNVi: Wi-Fi*	0 / 20 / 3	4D40h - 4D43h
IEH #0	0 / 20 / 5	4D44h
CNVI: Bluetooth*	0 / 20 / 7	4D76h
I²C Controller #0	0 / 21 / 0	4D78h
I²C Controller #1	0 / 21 / 1	4D79h
I²C Controller #2	0 / 21 / 2	4D7Ah
I²C Controller #3	0 / 21 / 3	4D7Bh
Intel® CSME: HECI #1 (CSE)	0 / 22 / 0	4D70h
Intel® CSME: HECI #2 (CSE)	0 / 22 / 1	4D71h
Intel® CSME: IDE Redirection (IDE-R)	0 / 22 / 2	4D72h
Intel® CSME: Keyboard and Text (KT) Redirection	0 / 22 / 3	4D73h
Intel® CSME: HECI #3 (CSE)	0 / 22 / 4	4D74h
Intel® CSME: HECI #4 (CSE)	0 / 22 / 5	4D75h
UFS Controller	0 / 23 / 0	4D47h
Intel® CSME: HECI #1	0 / 24 / 0	4D5Dh
Intel® CSME: HECI #2	0 / 24 / 1	4D5Eh
Intel® CSME: HECI #3	0 / 24 / 2	4D5Fh
I²C Controller #4	0 / 25 / 0	4D50h
I²C Controller #5	0 / 25 / 1	4D51h
UART #2	0 / 25 / 2	4D52h
PCI Express Root Port #1	0 / 28 / 0	4D3Ch

continued...

Device	Bus / Device / Function	6C Processor (DID)
PCI Express Root Port #2	0 / 28 / 1	4D3Dh
PCI Express Root Port #3	0 / 28 / 2	4D3Eh
PCI Express Root Port #4	0 / 28 / 3	4D3Fh
UART #0	0 / 30 / 0	4D25h
UART #1	0 / 30 / 1	4D26h
GSPI #0	0 / 30 / 2	4D27h
GSPI #1	0 / 30 / 3	4D30h
eSPI Controller	0 / 31 / 0	4D00h -4D1Fh
P2SB (8 bit)	0 / 31 / 1	4D20h
PMC	0 / 31 / 2	4D21h
Intel® High Definition Audio (Intel® HD Audio) AVS (Audio, Voice, Speech)	0 / 31 / 3	4D28h -4D2Fh
SMBus	0 / 31 / 4	4D22h
SPI (flash) Controller	0 / 31 / 5	4D23h
GbE Controller: Corporate/Intel® vPro™ (Default)	0 / 31 / 6	57B7h
GbE Controller: Consumer	0 / 31 / 6	57B8h
Intel® Trace Hub (Intel® TH)	0 / 31 / 7	4D24h

Table 9. ACPI Device ID for GPIO Controller

ACPI ID	Note
GPIO Controller	10ECh

2.4 Revision IDs

The Revision ID (RID) register is an 8-bit register located at offset 08h in the PCI header of every PCI/PCIe* function. The RID register is used by software to identify a particular component stepping when a driver change or patch unique to that stepping is needed.

The RID register reports one of the two possible values:

- Stepping Revision Identification (SRID)
- Compatible Revision ID (CRID)

The default power-on value for the RID register is SRID. The assigned value is based on the product's stepping. CRID is intended for the corporate Intel® Stable Image Platform Program (Intel® SIPP). CRID is normally identical to the SRID value of a previous production stepping of the product with which the new stepping is deemed compatible. Intel® SIPP allows an OS image built on the earlier stepping to be used on any new compatible stepping(s). Three CRID values are possible and can be used to manage software images.

NOTE

SRID and CRID are not addressable PCI registers. The SRID and CRID value are reflected through the RID register when appropriately selected.

Following reset, the SRID value can be read from the RID registers of all Processor devices and functions.

To select either SRID or CRID to be reflected in the RID registers:

1. BIOS needs to write appropriate value into the Configured Revision ID (CRID) register located in the PMC MMIO space.
2. BIOS must write this register with the appropriate value after S4/S5 states and after PLTRST# events.

After CRID is selected and applied by BIOS, software will not be able to obtain the original SRID value of the Processor by reading the RID registers. Customers implementing CRID who also want to determine the SRID in runtime may develop their own tool. For example, BIOS can capture the SRID value before BIOS applies CRID and store that value in a runtime accessible place (that is, SMBIOS, ACPI Type 4 Memory, NVRAM, CMOS) so that it can be read by the customer tool later. Alternatively, theBIOS can store the SRID value and display this information in BIOSsetup while reporting that CRID is enabled.

BIOS needs to check CRID_UIP bit (in PMC MMIO space) as a part of the update flow. PMC HW sets this bit to indicate that SetID broadcast flow has been requested by BIOS. This bit is cleared by PMC FW only when the completion/s of SetIDVal message is received by PMC. BIOS is required to read this bit as cleared before writing to the CRID register (to request a CRID update). BIOS is also required to poll on reads to this bit until it detects the bit as cleared after BIOS has written to the CRID register.

3.0 Package Mechanical Specifications

3.1 Package Mechanical Attributes

The Intel® Core™ Processor (Series 3) Processor Series use a Flip Chip technology available in a Ball Grid Array (BGA) package. The following table provides an overview of the package mechanical attributes. For specific dimensions (tile size, tile location, and so on).

Table 10. Intel® Core™ Processor (Series 3) Package Mechanical Attributes

Package Attributes	Parameter	Processor Series
Package Technology	Package Type	Flip Chip Ball Grid Array
	Interconnect	Ball Grid Array (BGA)
	Lead Free	Yes
	Halogenated Flame Retardant Free	Yes
Package Configuration	Solder Ball Composition	SAC
	Ball Count	1516
	Grid Array Pattern	Balls anywhere
	Land Side Capacitors	Yes
	Die Side Capacitors	No
Package Dimensions	Nominal Package Size	35 x 25 mm
	Tile Z-height (BGA Pre-SMT Bottom to Top) (mm)	1.055 ± 0.09
	Minimum Ball pitch	0.62 mm

3.2 Package Loading and Tile Pressure Specifications

Intel has defined the static compressive load limits and maximum pressure specs that can be applied to the package for the following SKUs.

3.2.1 Static Compressive Load Specification

3.2.2 Maximum Pressure Specifications

A more relevant metric for concentrated loading is chosen by Intel based on the physics of failure to evaluate tile damage risk due to thermal solution enabling .

- **Static Compressive Pressure** refers to the long-term steady state pressure applied to the tile from the thermal solution after system assembly is complete

- **Transient Compressive Pressure** refers to the pressure on the tiles at any moment during the thermal solution assembly/disassembly procedures. Other system procedures such as repair/rework can also cause high pressure loading to occur on the tile and should be evaluated to ensure these limits are not exceeded

Metric: This metric is pressure over a 2 mm x 2 mm area

Table 11. Package Loading Specifications

Static Compressive Pressure[1] [PSI]	Transient Compressive Pressure[1] [PSI]
800 psi	800 psi
Notes: • This is the load and pressure that has been tested by Intel for a single assembly cycle. This metric is a pressure over 2 mm^2 (2 mm x 2 mm) area. • For Static compressive pressure, the load conditions and corresponding pressure conditions need to be followed by the thermal attach design of maximum 10 lbf on 0.6 mm board and maximum of 15 lbf on 0.8 mm board.	

3.3 Package Storage Specifications

Parameter	Description	Minimum	Maximum
$T_{ABSOLUTE\ STORAGE}$	The non-operating device storage temperature. Damage (latent or otherwise) may occur when subjected to this temperature for any length of time in Intel Original sealed moisture barrier bag and / or box.	- 25°C	125°C
$T_{SUSTAINED\ STORAGE}$	The ambient storage temperature limit (in shipping media) for the sustained period of time	-5°C	40°C
$RH_{SUSTAINED\ STORAGE}$	The maximum device storage relative humidity for the sustained period of time as specified below in Intel Original sealed moisture barrier bag and / or box	60% @ 24°C	
$TIME_{SUSTAINED\ STORAGE}$	Maximum time: associated with customer shelf life in Intel Original sealed moisture barrier bag and / or box	NA	**Moisture Sensitive Devices:** 60 months from bag seal date; **Non-moisture sensitive devices:** 60 months from lot date
Storage Conditions	Processors in a non-operational state may be installed in a platform, in a tray, boxed, or loose and may be sealed in airtight package 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 should be handled in accordance with moisture sensitivity labeling (MSL) as indicated on the packaging material. Boxed Land Grid Array packaged (LGA) processors are MSL 1 ('unlimited' or unaffected) as they are not heated in order to be inserted in the socket.		
Notes: 1. $T_{ABSOLUTE\ STORAGE}$ applies to the un-assembled component only and does not apply to the shipping media, moisture barrier bags or desiccant. Refers to a component device that is not assembled in a board or socket that is not to be electrically connected to a voltage reference or I/O signals. 2. Specified temperatures are based on data collected. Exceptions for surface mount re-flow are specified by applicable JEDEC J-STD-020 and MAS documents. The JEDEC, J-STD-020 moisture level rating and associated handling practices apply to all moisture sensitive devices removed from the moisture barrier bag. 3. Post board attaches storage temperature limits are not specified for non-Intel branded boards. Consult your board manufacturer for storage specifications.			

4.0 Memory Mapping

This chapter describes (from the processor perspective) the memory ranges that the Processor decodes.

4.1 Functional Description

4.1.1 PCI Devices and Functions

The Intel® Core™ Series 3 Processor incorporates a variety of PCI devices and functions, as shown in the following table. If for some reason, the particular system platform does not want to support any one of the Device Functions, with the exception of D30:F0, they can individually be disabled. The integrated Gigabit Ethernet controller will be disabled if no Platform LAN Connect component is detected (Gigabit Ethernet Controller on page 192). When a function is disabled, it does not appear to the software. A disabled function will not respond to any register reads or writes, ensuring that these devices appear hidden to software.

4.1.2 Fixed IO Address Ranges

The following table shows the Fixed I/O decode ranges from the processor perspective.

NOTE

For each I/O range, there may be separate behavior for reads and writes.

I/O cycles that go to target ranges that are marked as Reserved will be handled as follows: writes are ignored and reads will return all 1's. The P2SB will claim many of the fixed I/O accesses and forward those transactions over IOSF-SB to their functional target.

Address ranges that are not listed or marked Reserved are NOT positively decoded (unless assigned to one of the variable ranges) and will be internally terminated.

Table 12. Fixed I/O Ranges Decoded by Processor

I/O Address	Read Target	Write Target	Internal Unit (Unless[E]: External)[2]	Separate Enable/Disable
20h – 21h	Interrupt Controller	Interrupt Controller	Interrupt	None
24h – 25h	Interrupt Controller	Interrupt Controller	Interrupt	None
28h – 29h	Interrupt Controller	Interrupt Controller	Interrupt	None
2Ch – 2Dh	Interrupt Controller	Interrupt Controller	Interrupt	None
2E-2F	Super I/O	Super I/O	[E] Forwarded to eSPI	Yes. ESPI_IOD_IOE.SE

continued...

I/O Address	Read Target	Write Target	Internal Unit (Unless[E]: External)[2]	Separate Enable/Disable
30h – 31h	Interrupt Controller	Interrupt Controller	Interrupt	None
34h – 35h	Interrupt Controller	Interrupt Controller	Interrupt	None
38h – 39h	Interrupt Controller	Interrupt Controller	Interrupt	None
3Ch – 3Dh	Interrupt Controller	Interrupt Controller	Interrupt	None
40h	Timer/Counter	Timer/Counter	8254 Timer	None
42h-43h	Timer/Counter	Timer/Counter	8254 Timer	None
4E-4F	Microcontroller	Microcontroller	[E] Forwarded to eSPI	Yes. ESPI_IOD_IOE.ME2
50h	Timer/Counter	Timer/Counter	8254 Timer	None
52h-53h	Timer/Counter	Timer/Counter	8254 Timer	None
60h	Keyboard Controller	Keyboard Controller	[E] Forwarded to eSPI	Yes, with 64h. ESPI_IOD_IOE.KE
61h	NMI Controller	NMI Controller	Processor I/F	None
62h	Microcontroller	Microcontroller	[E] Forwarded to eSPI	Yes, with 66h. ESPI_IOD_IOE.ME1
63h	NMI Controller [1]	NMI Controller [1]	Processor I/F	Yes, alias to 61h. GIC.P61AE
64h	Keyboard Controller	Keyboard Controller	[E] Forwarded to eSPI	Yes, with 60h. ESPI_IOD_IOE.KE
65h	NMI Controller [1]	NMI Controller [1]	Processor I/F	Yes, alias to 61h. GIC.P61AE
66h	Microcontroller	Microcontroller	[E] Forwarded to eSPI	Yes, with 62h. ESPI_IOD_IOE.ME1
67h	NMI Controller [1]	NMI Controller [1]	Processor I/F	Yes, alias to 61h. GIC.P61AE
70h	RTC Controller	NMI and RTC Controller	RTC	None
71h	RTC Controller	RTC Controller	RTC	None
72h	RTC Controller	RTC Controller	RTC	None. Alias to 70h if RC.UE[4]=0, else 72h
73h	RTC Controller	RTC Controller	RTC	None. Alias to 71h if RC.UE='0', else 73h
74h	RTC Controller	RTC Controller	RTC	None
75h	RTC Controller	RTC Controller	RTC	None
76h-77h	RTC Controller	RTC Controller	RTC	None. Alias to 70h-71h if RC.UE=0, else 76h-77h
80h[3]	eSPI or PCIe	eSPI or PCIe	Read: [E] eSPI or PCIe	None. PCIe if GCS.RPR='1',

continued...

I/O Address	Read Target	Write Target	Internal Unit (Unless[E]: External)[2]	Separate Enable/Disable
			Write: [E] eSPI or [E] PCIe	else eSPI
84h - 86h	eSPI or PCIe	eSPI or PCIe	Read: [E] eSPI or PCIe Write: [E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else eSPI
88h	eSPI or PCIe	eSPI or PCIe	Read: [E] eSPI or PCIe Write: [E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else eSPI
8Ch - 8Eh	eSPI or PCIe	eSPI or PCIe	Read: [E] eSPI or PCIe Write: [E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else eSPI
90h	eSPI	eSPI	Read: [E] eSPI Write: [E] eSPI	None. Alias to 80h
92h	Reset Generator	Reset Generator	Processor I/F	None
94h - 96h	eSPI	eSPI	Read: [E] eSPI Write: [E] eSPI	None. Alias to 8xh
98h	eSPI	eSPI	Read: [E] eSPI Write: [E] eSPI	None. Alias to 88h
9Ch - 9Eh	eSPI	eSPI	Read: [E] eSPI Write: [E] eSPI	None. Alias to 8xh
A0h - A1h	Interrupt Controller	Interrupt Controller	Interrupt	None
A4h - A5h	Interrupt Controller	Interrupt Controller	Interrupt	None
A8h - A9h	Interrupt Controller	Interrupt Controller	Interrupt	None
ACh - ADh	Interrupt Controller	Interrupt Controller	Interrupt	None
B0h - B1h	Interrupt Controller	Interrupt Controller	Interrupt	None
B2h - B3h	Power Management	Power Management	Power Management	None
B4h - B5h	Interrupt Controller	Interrupt Controller	Interrupt	None
B8h - B9h	Interrupt Controller	Interrupt Controller	Interrupt	None
BCh - BDh	Interrupt Controller	Interrupt Controller	Interrupt	None

continued...

I/O Address	Read Target	Write Target	Internal Unit (Unless[E]: External)[2]	Separate Enable/Disable
200-207h	Gameport Low	Gameport Low	Forwarded to eSPI	Yes. ESPI_CS1IORE.LGE
208-20Fh	Gameport High	Gameport High	Forwarded to eSPI	Yes ESPI_CS1IORE.HGRE
4D0h – 4D1h	Interrupt Controller	Interrupt Controller	Interrupt Controller	None
CF9h	Reset Generator	Reset Generator	Interrupt controller	None

Notes: 1. Only if the Port 61 Alias Enable bit (GIC.P61AE) bit is set. Otherwise, the cycle is internally terminated by the Processor.
2. Destination of eSPI when eSPI Disabled pin strap is 0.
3. This includes byte, word, or double-word (DW) access at I/O address 80h.

4.1.3 Variable IO Decode Ranges

The following Table shows the Variable I/O Decode Ranges. They are set using Base Address Registers (BARs) or other configuration bits in the various configuration spaces. The PnP software (PCI or ACPI) can use their configuration mechanisms to set and adjust these values.

WARNING

The Variable I/O Ranges should not be set to conflict with the Fixed I/O Ranges. There may some unpredictable results if the configuration software allows conflicts to occur. The Processor does not perform any checks for conflicts.

Table 13. Variable I/O Decode Ranges

Range Name[1]	Mappable	Size (Bytes)	Target
ACPI	Anywhere in 64K I/O Space	256	Power Management
IDE Bus Host	Anywhere in 64K I/O Space	16 or 32 Bytes	Intel® AMT IDE-R
SMBus	Anywhere in 64K I/O Space	32	SMB Unit
TCO	Anywhere in 64K I/O Space	32	SMB Unit
Parallel Port	3 ranges in 64K I/O Space	8	eSPI
Serial Port 1	8 Ranges in 64K I/O Space	8	eSPI
Serial Port 2	8 Ranges in 64K I/O Space	8	eSPI
Serial Port 3	8 Ranges in 64K I/O space	8	eSPI
LPC Generic 1	Anywhere in 64K I/O Space	4 to 256 Bytes	eSPI
LPC Generic 2	Anywhere in 64K I/O Space	4 to 256 Bytes	eSPI
LPC Generic 3	Anywhere in 64K I/O Space	4 to 256 Bytes	eSPI
LPC Generic 4	Anywhere in 64K I/O Space	4 to 256 Bytes	eSPI
IO Trapping Ranges	Anywhere in 64K I/O Space	1 to 256 Bytes	Trap

continued...

Range Name[1]	Mappable	Size (Bytes)	Target
PCI Express* Root Ports	Anywhere in 64K I/O Space	I/O Base/Limit	PCI Express* Root Ports 1-28
Keyboard and Text (KT)	Anywhere in 64K I/O Space	8	Intel® AMT Keyboard and Text
Note: All ranges are decoded directly from IOC.			

4.2 Memory Map

The following table shows (from the processor perspective) the memory ranges that the processor will decode. Cycles that are not directed to any of the internal memory targets, will be host aborted.

PCIe cycles generated by external PCIe hosts will be positively decoded unless they fall in the PCI-PCI bridge memory forwarding ranges (those addresses are reserved for PCI peer-to-peer traffic). Software must not attempt locks to the processor's memory-mapped I/O ranges.

NOTE

Total ports are different for the different SKUs.

Table 14. Processor Memory Decode Ranges (Processor Perspective)

Memory Range	Target	Dependency/Comments
000E 0000 - 000E FFFF	eSPI or SPI	Bit 6 in BIOS Decode Enable Register is set
000F 0000 - 000F FFFF	eSPI or SPI	Bit 7 in BIOS Decode Enable Register is set
FECX X000 - FECX X040	I/O(x)APIC inside processor	XX controlled via APIC Range Select (ASEL) field and APIC Enable (AEN) bit
FECX X000 - FECX XFFF	PCIe port N (N=1 to 28)	X controlled via PCIe root port N IOxAPIC Range Base/Limit registers and Port N I/OxApic Enable (PAE) is set
FEC1 0000 - FEC1 7FFF	PCIe port 1	PCIe root port 1 I/OxApic Enable (PAE) is set
FEC1 8000 - FEC1 FFFF	PCIe port 2	PCIe root port 2 I/OxApic Enable (PAE) is set
FEC2 0000 - FEC2 7FFF	PCIe port 3	PCIe root port 3 I/OxApic Enable (PAE) is set
FEC2 8000 - FEC2 FFFF	PCIe port 4	PCIe root port 4 I/OxApic Enable (PAE) is set
FEC3 0000 - FEC3 7FFF	PCIe port 5	PCIe root port 5 I/OxApic Enable (PAE) is set
FEC3 8000 - FEC3 FFFF	PCIe port 6	PCIe root port 6 I/OxApic Enable (PAE) is set
FEC4 0000 - FEC4 7FFF	PCIe port 7	PCIe root port 7 I/OxApic Enable (PAE) is set
FEC4 8000 - FEC4 FFFF	PCIe port 8	PCIe root port 8 I/OxApic Enable (PAE) is set
FEC5 0000 - FEC5 7FFF	PCIe port 9	PCIe root port 9 I/OxApic Enable (PAE) is set
FEC5 8000 - FEC5 FFFF	PCIe port 10	PCIe root port 10 I/OxApic Enable (PAE) is set
FEC6 0000 - FEC6 7FFF	PCIe port 11	PCIe root port 11 I/OxApic Enable (PAE) is set
FEC6 8000 - FEC6 FFFF	PCIe port 12	PCIe root port 12 I/OxApic Enable (PAE) is set
FEC7 0000 - FEC7 7FFF	PCIe port 13	PCIe root port 13 I/OxApic Enable (PAE) is set
FEC7 8000 - FEC7 FFFF	PCIe port 14	PCIe root port 14 I/OxApic Enable (PAE) is set

continued...

Memory Range	Target	Dependency/Comments
FEC8 0000 - FEC8 7FFF	PCIe port 15	PCIe root port 15 I/OxApic Enable (PAE) is set
FEC8 8000 - FEC8 FFFF	PCIe port 16	PCIe root port 16 I/OxApic Enable (PAE) is set
FEC9 0000 - FEC9 7FFF	PCIe port 17	PCIe root port 17 I/OxApic Enable (PAE) is set
FEC9 8000 - FEC9 FFFF	PCIe port 18	PCIe root port 18 I/OxApic Enable (PAE) is set
FECA 0000 - FECA 7FFF	PCIe port 19	PCIe root port 19 I/OxApic Enable (PAE) is set
FECA 8000 - FECA FFFF	PCIe port 20	PCIe root port 20 I/OxApic Enable (PAE) is set
FECB 0000 - FECB 7FFF	PCIe port 21	PCIe root port 21 I/OxApic Enable (PAE) is set
FECB 8000 - FECB FFFF	PCIe port 22	PCIe root port 22 I/OxApic Enable (PAE) is set
FECC 0000 - FECC 7FFF	PCIe port 23	PCIe root port 23 I/OxApic Enable (PAE) is set
FECC 8000 - FECC FFFF	PCIe port 24	PCIe root port 24 I/OxApic Enable (PAE) is set
FECD 0000 - FECD 7FFF	PCIe port 25	PCIe root port 25 I/OxApic Enable (PAE) is set
FECD 8000 - FECD FFFF	PCIe port 26	PCIe root port 26 I/OxApic Enable (PAE) is set
FECE 0000 - FECE_7FFF	PCIe port 27	PCIe root port 27 I/OxApic Enable (PAE) is set
FECE 8000 - FECE FFFF	PCIe port 28	PCIe root port 28 I/OxApic Enable (PAE) is set
FEF0 0000 - FEFF FFFF	eSPI or SPI	uCode Patch Region Enable UCPR.UPRE is set
FFC0 0000 - FFC7 FFFF FF80 0000 - FF87 FFFF	eSPI or SPI	Bit 8 in BIOS Decode Enable Register is set
FFC8 0000 – FFCF FFFF FF88 0000 - FF8F FFFF	eSPI or SPI	Bit 9 in BIOS Decode Enable Register is set
FFD0 0000 - FFD7 FFFF FF90 0000 - FF97 FFFF	eSPI or SPI	Bit 10 in BIOS Decode Enable Register is set
FFD8 0000 – FFDF FFFF FF98 0000 - FF9F FFFF	eSPI or SPI	Bit 11 in BIOS Decode Enable Register is set
FFE0 0000 - FFE7 FFFF FFA0 0000 - FFA7 FFFF	eSPI or SPI	Bit 12 in BIOS Decode Enable Register is set
FFE8 0000 – FFEF FFFF FFA8 0000 – FFAF FFFF	eSPI or SPI	Bit 13 in BIOS Decode Enable Register is set
FFF0 0000 - FFF7 FFFF FFB0 0000 - FFB7 FFFF	eSPI or SPI	Bit 14 in BIOS Decode Enable Register is set
FFFC 0000 - FFFF FFFF	eSPI, SPI, or Intel® CSME	Always enabled. Refer to Table 15 on page 40 for swappable ranges
FFF8 0000 - FFFB FFFF FFB8 0000 - FFBF FFFF	eSPI or SPI	Always enabled. Refer to Table 15 on page 40 for swappable ranges
FF70 0000 - FF7F FFFF FF30 0000 - FF3F FFFF	eSPI or SPI	Bit 3 in BIOS Decode Enable Register is set
FF60 0000 - FF6F FFFF FF20 0000 - FF2F FFFF	eSPI or SPI	Bit 2 in BIOS Decode Enable Register is set
FF50 0000 - FF5F FFFF FF10 0000 - FF1F FFFF	eSPI or SPI	Bit 1 in BIOS Decode Enable Register is set
FF40 0000 - FF4F FFFF	eSPI or SPI	Bit 0 in BIOS Decode Enable Register is set

continued...

Memory Range	Target	Dependency/Comments
FF00 0000 - FF0F FFFF		
FED0 X000 - FED0 X3FF	HPET	BIOS determines "fixed" location which is one of four 1 KB ranges where X (in the first column) is 0h, 1h, 2h, or 3h
FED4 0000 - FED4 7FFF	SPI (set by strap)	TPM and Trusted Mobile KBC
FED4 C000 - FED4 FFFF	Processor Internal (PSF Error Handler)	Always enabled
FED6 0000 – FED6 1FFF	Processor Internal (Intel® Trace Hub (Intel® TH)/xHCI)	Always enabled
FED5 0000 - FED5 FFFF	Intel® CSME	Always enabled
FED7 0000 - FED7 4FFF	Internal Device	Security feature related
128 KB anywhere in 4 GB range	LAN Controller (CSR registers)	Enable via standard PCI mechanism (Device 31:Function 6)
4 KB anywhere in 4 GB range	LAN Controller (LAN space on Flash)	Enable via standard PCI mechanism (Device 31:Function 6)
64 KB anywhere in 64-bit address range	USB Host Controller	Enable via standard PCI mechanism (Device 20, Function 0)
2 MB anywhere in 4 GB range	USB Device Controller	Enable via standard PCI mechanism (Device 20, Function 1)
24 KB anywhere in 4 GB range	USB Device Controller	Enable via standard PCI mechanism (Device 20, Function 1)
16 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
4 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
64 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
32 Bytes anywhere in 64-bit address range	SMBus	Enable via standard PCI mechanism (Device 31: Function 4)
Memory Base/Limit anywhere in 4 GB range	PCI Express* Root Ports 1-28	Enable via standard PCI mechanism
Prefetchable Memory Base/Limit anywhere in 64-bit address range	PCI Express* Root Ports 1-28	Enable via standard PCI mechanism
16 Bytes anywhere in 64-bit address range	Intel® CSMEI #1, #2, #3, #4	Enable via standard PCI mechanism
4 KB anywhere in 4 GB range	Intel® AMT Keyboard and Text	Enable via standard PCI mechanism (Device 22: Function 3)
16 MB anywhere in 64-bit address range	P2SB	Enable via standard PCI mechanism
12 4 KB slots anywhere in 64-bit address range	I^3C function has 8 KB BAR, all others (I^2C/SPI/UART) are 4 KB.	Enable via standard PCI mechanism
1 MB (BAR0) or 4 KB (BAR1) in 4GB range	Integrated Sensor Hub	Enable via standard PCI mechanism (Device 19: Function 0)
8 KB slot anywhere in 4 GB range	Integrated Wi-Fi*	Enable via standard PCI mechanism

continued...

Memory Range	Target	Dependency/Comments
8 KB slot and 4 KB slot anywhere in 4 GB range	PMC	Enable via standard PCI mechanism
8 KB slot and 4 KB slot anywhere in 4 GB range	Shared SRAM	Enable via standard PCI mechanism
Two 32 KB anywhere in 64-bit address range	THC #0, #1	Enable via standard PCI mechanism

4.2.1 Boot Block Update Scheme

The Processor supports a "Top-Block Swap" mode that has the Processor swap the top block in the SPI flash (the boot block) with another location. This allows for safe update of the Boot Block (even if a power failure occurs). When the "top-swap" enable bit is set, the Processor will invert A16 for cycles going to the upper two 64-KB blocks in the appropriate address lines as selected in Boot Block Size (BOOT_BLOCK_SIZE) soft strap for SPI.

For SPI when top swap is enabled, the behavior is as described below. When the Top Swap Enable bit is 0, the Processor will not invert any address bit.

Table 15. Boot Block Update Scheme

BOOT_BLOCK_SIZE Value	Accesses to	Being Directed to
000 (64KB)	FFFF_0000h - FFFF_FFFFh	FFFE_0000h - FFFE_FFFFh and vice versa
001 (128KB)	FFFE_0000h - FFFF_FFFFh	FFFC_0000h - FFFD_FFFFh and vice versa
010 (256KB)	FFFC_0000h - FFFF_FFFFh	FFF8_0000h - FFFB_FFFFh and vice versa
011 (512KB)	FFF8_0000h - FFFF_FFFFh	FFF0_0000h - FFF7_FFFFh and vice versa
100 (1MB)	FFF0_0000h - FFFF_FFFFh	FFE0_0000h - FFEF_FFFFh and vice versa
101 - 111	Reserved	Reserved
Note: This bit is automatically set to 0 by RTCRST#, but not by PLTRST#.		

The scheme is based on the concept that the top block is reserved as the "boot" block, and the block immediately below the top block is reserved for doing boot-block updates.

The algorithm is:

1. Software copies the top block to the block immediately below the top
2. Software checks that the copied block is correct. This could be done by performing a checksum calculation.
3. Software sets the "Top-Block Swap" bit. This will invert the appropriate address bits for the cycles going to the SPI.
4. Software erases the top block
5. Software writes the new top block
6. Software checks the new top block
7. Software clears the top-block swap bit
8. Software sets the Top_Swap Lock-Down bit

If a power failure occurs at any point after step 3, the system will be able to boot from the copy of the boot block that is stored in the block below the top. This is because the top-swap bit is backed in the RTC well.

There is one remaining unusual case that could occur if the RTC battery voltage is not sufficiently high to maintain the RTC well. To avoid the potentially fatal case (where the Top-Swap bit is NOT set, but the top block is not valid), a pin strap will allow forcing the top-swap bit to be set. This would be a last resort to allow the user to get the system to boot (and avoid having to de-solder the system flash).

When the top-swap strap is used, the top-swap bit will be forced to 1 (cannot be cleared by software).

The BIOS algorithm should be as follows:

1. If an RTC well power failure is experienced during a boot block update, the system will probably not be able to boot at that point.
2. The user can set the Top-Swap pin strap and force the system to boot from the 2nd block. The code in the 2nd block should read the valid BIOS image from disk and put it into the top-swap.
3. The BIOS will not clear the Top-Swap bit (because the jumper is in place). The user should then remove the jumper and reboot.

5.0 Security Technologies

5.1 Intel® Converged Boot Guard and TXT

Intel® Converged Boot Guard and Intel® TXT (Intel® CBnT) is an unification of Intel® Trusted Execution Technology (Intel® TXT) and Intel® Platform Protection Technology with Intel® Boot Guard. Intel® CBnT merges elements of Intel® TXT and Intel® Boot Guard to enhance platform boot security, while also simplifying the implementation. Although Intel® CBnT implements some architectural changes, it is not fundamentally a new technology, but rather a fusion of existing Intel® Boot Guard and Intel® TXT technologies.

Intel® CBnT has been designed to allow greater commonality between implementations for client platforms and server platforms. Previously, the architectural implementation of Intel® TXT was somewhat different between client and server platforms, which necessitated some differences in BIOS implementation depending on the platform. With Intel® CBnT, Intel has largely combined features across client and server providing greater alignment in design of the BIOS and ACMs.

Intel® Converged Boot Guard and Intel® TXT provides both a static root of trust for verifying the BIOS initial boot block and measuring the boot path, as well as a dynamic root of trust for measuring the OS or VMM.

The purpose of Intel® Boot Guard is to verify that the initial BIOS startup code is good, i.e., BIOS has not been maliciously nor inadvertently modified. Several different Boot Profiles are supported, which primarily differ in:

- **Enforcement Policy**: what actions are taken if BIOS cannot be verified.
- **Measurement Policy**: whether BIOS startup code is measured into the TPM for attestation.

The primary objective of Intel® TXT is to provide a dynamic root of trust for measuring the OS or VMM enabling platform boot into a secure measured launch environment (MLE). Intel® TXT relies on the static root of trust provided by Intel® Boot Guard to ensure validity of the MLE Trusted Compute Base (TCB), which is the BIOS code that is trusted to configure the platform. Intel® TXT provides the ability to allow only a known good OS/VMM to launch into a trusted environment via a Launch Control Policy (LCP). And once an OS/VMM is in a trusted environment, Intel® TXT protects memory secrets against surprise reset attacks.

With the modifications made to the Intel® TXT architecture in Intel® CBnT, it is now required that some of the verifications performed by Intel® Boot Guard be implemented for Intel® TXT support. Verifications of pre-boot objects such as FIT, key and policy manifests, and of Startup BIOS.

Still formally all four combinations of constituent technologies are supported at OEM choice:

- Intel® Boot Guard only enabled.
- Intel® TXT only enabled.

- Both Intel® Boot Guard and Intel® TXT enabled.

5.2 Crypto Acceleration Instructions

5.2.1 Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)

The processor supports Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI) that are a set of Single Instruction Multiple Data (SIMD) instructions that enable fast and secure data encryption and decryption based on the Advanced Encryption Standard (AES). Intel® AES-NI is valuable for a wide range of cryptographic applications, such as applications that perform bulk encryption/ decryption, authentication, random number generation, and authenticated encryption. AES is broadly accepted as the standard for both government and industrial applications and is widely deployed in various protocols.

Intel® AES-NI consists of six Intel® Streaming SIMD Extensions (Intel® SSE) instructions. Four instructions, AESENC, AESENCLAST, AESDEC, and AESDELAST facilitate high-performance AES encryption and decryption. The other two, AESIMC and AESKEYGENASSIST, support the AES key expansion procedure. Together, these instructions provide full hardware for supporting AES; offering security, high performance, and a great deal of flexibility.

The Intel® AES-NI specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 2*. Available at:

http://www.intel.com/products/processor/manuals

NOTE

Intel® AES-NI Technology may not be available on all SKUs.

5.2.2 Perform Carry-Less Multiplication Quad Word Instruction (PCLMULQDQ)

The processor supports the carry-less multiplication instruction, PCLMULQDQ. PCLMULQDQ is a Single Instruction Multiple Data (SIMD) instruction that computes the 128-bit carry-less multiplication of two 64-bit operands without generating and propagating carries. Carry-less multiplication is an essential processing component of several cryptographic systems and standards. Hence, accelerating carry-less multiplication can significantly contribute to achieving high-speed secure computing and communication.

PCLMULQDQ specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 2*. Available at:

http://www.intel.com/products/processor/manuals

5.2.3 Intel® Secure Hash Algorithm Extensions (Intel® SHA Extensions)

The Secure Hash Algorithm (SHA) is one of the most commonly employed cryptographic algorithms. Primary usages of SHA include data integrity, message authentication, digital signatures, and data de-duplication. As the pervasive use of security solutions continues to grow, SHA can be seen in more applications now than ever. The Intel® SHA Extensions are designed to improve the performance of these compute-intensive algorithms on Intel® architecture-based processors.

The Intel® SHA Extensions are a family of seven instructions based on the Intel® Streaming SIMD Extensions (Intel® SSE) that are used together to accelerate the performance of processing SHA-1 and SHA-256 on Intel architecture-based processors. Given the growing importance of SHA in our everyday computing devices, the instructions are designed to provide a needed boost of performance to hashing a single buffer of data. The performance benefits will not only help improve responsiveness and lower power consumption for a given application, but they may also enable developers to adopt SHA in new applications to protect data while delivering to their user experience goals. The instructions are defined in a way that simplifies their mapping into the algorithm processing flow of most software libraries, thus enabling easier development.

Information on Intel® SHA can be found at: http://software.intel.com/en-us/artTGLes/intel-sha-extensions

5.2.4 New Cryptographic Acceleration Instructions

The processor supports new extensions for acceleration of some common or emerging cryptographic algorithms:

1. AVX2 version of VPMADD52 for acceleration of RSA signature verification
2. SHA2-512 (or 384)
3. Chinese crypto standards SM3 and SM4

5.3 Intel® Secure Key

The processor supports Intel® Secure Key (formerly known as Digital Random Number Generator or DRNG), a software visible random number generation mechanism supported by a high-quality entropy source. This capability is available to programmers through the RDRAND and RDSEED instructions. The resultant random number generation capability is designed to comply with existing industry standards in this regard (ANSI X9.82 and NIST SP 800-90).

Some possible usages of the RDRAND and RDSEED instructions include cryptographic key generation as used in a variety of applications, including communication, digital signatures, secure storage, etc.

RDRAND and RDSEED instructions specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 2*. Available at:

http://www.intel.com/products/processor/manuals

5.4 Execute Disable Bit

The Execute Disable Bit allows memory to be marked as 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, thus, help improve the overall security of the system.

5.5 Intel® Supervisor Mode Execution Protection (Intel® SMEP)

Intel® Supervisor Mode Execution Protection (Intel® SMEP) is a mechanism that provides the next level of system protection by blocking malicious software attacks from user mode code when the system is running in the highest privilege level. This technology helps to protect from virus attacks and unwanted code from harming the system. For more information, refer to *Intel ® 64 Architectures Software Developer's Manual, Volume 3* at:

http://www.intel.com/products/processor/manuals

5.6 Intel® Supervisor Mode Access Protection (Intel® SMAP)

Intel® Supervisor Mode Access Protection (Intel® SMAP) is a mechanism that provides next level of system protection by blocking a malicious user from tricking the operating system into branching off user data. This technology shuts down very popular attack vectors against operating systems.

For more information, refer to Intel® *64 Architectures Software Developer's Manual, Volume 3*:

http://www.intel.com/products/processor/manuals

5.7 User Mode Instruction Prevention (UMIP)

User Mode Instruction Prevention (UMIP) provides additional hardening capability to the OS kernel by allowing certain instructions to execute only in supervisor mode (Ring 0).

If the OS opt-in to use UMIP, the following instruction are enforced to run in supervisor mode:

- **SGDT** - Store the GDTR register value
- **SIDT** - Store the IDTR register value
- **SLDT** - Store the LDTR register value
- **SMSW** - Store Machine Status Word
- **STR** - Store the TR register value

An attempt at such execution in user mode causes a general protection exception (#GP).

UMIP specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 3*. Available at:

http://www.intel.com/products/processor/manuals

5.8 Read Processor ID (RDPID)

A companion instruction that returns the current logical processor's ID and provides a faster alternative to using the RDTSCP instruction.

RDPID specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 2*. Available at:

http://www.intel.com/products/processor/manuals

5.9 Intel® System Resources Defense and Intel® System Security Report

Intel® System Resources Defense is the collection of techniques and code within the BIOS used to create and enforce HW access policy for the SMI handler. It consists of a collection of policy mechanisms that are configured by POST before the SMI handler is locked down. Once the SMI handler is locked, all accesses into the system must be compliant with the policy established during POST.

Intel® Runtime BIOS Resilience is a subset of Intel® System Resources Defense covering SMM memory policy only. Intel® Runtime BIOS Resilience Protection hardens the SMI handler via hardware enforced BIOS policy regarding SMI handler access to memory using an enhanced paging policy. This paging policy covers SMI handler access to both BIOS and MLE resources. Intel® Runtime BIOS Resilience Protection is extended using a technology codenamed Intel® System Security Report.

The Platform Properties Assessment Module (PPAM) is the primary component of Intel® System Security Report. It collects and reports information about platform SMM implementation and configuration, in order to provide trustworthy attestation of the resulting SMI memory policy regarding SMM secure configuration and access to MLE owned memory. Intel® System Security Report is used to create a trustworthy report describing the SMM policy. PPAM is a major/core component of Intel® System Security Report 1.0/1.1 technology

5.10 Intel® Total Memory Encryption - Multi-Key

This technology encrypts the platform's entire memory with multiple encryption keys. Intel® Total Memory Encryption (Intel® TME), when enabled via BIOS configuration, ensures that all memory accessed from the Intel processor is encrypted.

Intel TME encrypts memory accesses using the AES XTS algorithm with 256-bit keys. The global encryption key used for memory encryption is generated using a hardened random number generator in the processor and is not exposed to software.

Software (OS/VMM) manages the use of keys and can use each of the available keys for encrypting any page of the memory. Thus, Intel® Multi-key Total Memory Encryption (Intel® MK-TME) allows page granular encryption of memory. By default Intel MK-TME uses the Intel TME encryption key unless explicitly specified by software.

Data in-memory and on the external memory buses is encrypted and exists in plain text only inside the processor. This allows existing software to operate without any modification while protecting memory using Intel TME. Intel TME does not protect memory from modifications.

Intel TME allows the BIOS to specify a physical address range to remain unencrypted. Software running on Intel TME enabled system has full visibility into all portions of memory that are configured to be unencrypted by reading a configuration register in the processor.

NOTES

- Memory access to nonvolatile memory (Intel® Optane™) is encrypted as well.
- More information on Intel MK-TME can be found at:

  https://software.intel.com/sites/default/files/managed/a5/16/Total-Memory-Encryption-Multi-Key-Spec.pdf
- A cold boot is required when enable/ disable Intel TME feature on this platform.

5.11 Control-flow Enforcement Technology (Intel® CET)

Return-oriented Programming (ROP), and similarly CALL/JMP-oriented programming (COP/JOP), have been the prevalent attack methodology for stealth exploit writers targeting vulnerabilities in programs.

Intel® CET provides the following components to defend against ROP/JOP style control-flow subversion attacks:

5.11.1 Shadow Stack

A shadow stack is a second stack for the program that is used exclusively for control transfer operations. This stack is separate from the data stack and can be enabled for operation individually in user mode or supervisor mode.

The shadow stack is protected from tamper through the page table protections such that regular store instructions cannot modify the contents of the shadow stack. To provide this protection the page table protections are extended to support an additional attribute for pages to mark them as “Shadow Stack” pages. When shadow stacks are enabled, control transfer instructions/flows such as near call, far call, call to interrupt/exception handlers, etc. store their return addresses to the shadow stack. The RET instruction pops the return address from both stacks and compares them. If the return addresses from the two stacks do not match, the processor signals a control protection exception (#CP). Stores from instructions such as MOV, XSAVE, etc. are not allowed to the shadow stack.

5.11.2 Indirect Branch Tracking

The ENDBR32 and ENDBR64 (collectively ENDBRANCH) are two instructions that are used to mark valid indirect CALL/JMP target locations in the program. This instruction is a NOP on legacy processors for backward compatibility.

The processor implements a state machine that tracks indirect JMP and CALL instructions. When one of these instructions is seen, the state machine moves from IDLE to WAIT_FOR_ENDBRANCH state. In WAIT_FOR_ENDBRANCH state the next instruction in the program stream must be an ENDBRANCH. If an ENDBRANCH is not seen the processor causes a control protection exception (#CP), otherwise the state machine moves back to IDLE state.

More information on Intel® CET can be found at Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1, Chapter 18:

https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

5.12 BIOS Guard

The platform must implement hardware controls to provide the platform manufacturer a robust mechanism to prevent unauthorized flash updates, while still allowing platform manufacturer approved updates. Intel® Platform Protection Technology with Intel® BIOS Guard accomplishes this by providing a very robust environment from which signed update images can be cryptographically verified and host flash writes can be done. Furthermore, a Intel® BIOS Guard enabled system does not allow host flash writes from any other environment.

5.13 Intel® Platform Trust Technology

Intel® Platform Trust Technology (Intel® PTT) offers the capabilities of discrete TPM 2.0. Intel PTT is a platform functionality for credential storage and key management used by Windows* 8 , Windows* 10 and Windows* 11. Intel PTT supports BitLocker* for hard drive encryption and supports all Microsoft* requirements for Trusted Platform Module (TPM) 2.0.

5.14 Linear Address Space Separation (LASS)

Linear Address Space Separation (LASS) can harden an OS kernel against specific classes of side channel exploit techniques.

5.15 Intel® Total Storage Encryption (Intel® TSE)

Intel® Total Storage Encryption provides a security measure for a PCIe-NVMe device by encrypting the data in the device using the Intel inline encryption.

The Intel® Total Storage Encryption, which is enabled over BIOS configuration, ensures that the data in the storage device and through external PCIe buses are encrypted and exist in plaintext only inside the Intel processor.

The Intel® Total Storage Encryption encrypts the data using the AES XTS algorithm with a 256-bit data encryption key and 256-bit tweak key when the data is written to the storage from system memory and decrypted when read from the storage to the system memory.

The software can wrap the 256-bit keys using Platform Bind Key BLOB (PBNDKB) instruction and get the key handle wrapped by a platform-specific wrapping key. Once the software obtains the handle, the software can delete the original keys from memory. The software can also program the 256-bit keys using either the key handles or plaintext keys through the Platform Config (PCONFIG) instruction.

The Intel® Total Storage Encryption driver or the UEFI Inline Cryptographic Interface Protocol programs the keys over the PBNDKB/PCONFIG instructions and creates the Intel® Total Storage Encryption table in the system memory so that the Intel® Total Storage Encryption HW can look up the table to identify the Key ID with tweak value based on the physical address accessed through the storage driver of the PCIe-NVMe

device. BitLocker could use the Intel® Total Storage Encryption as security enhancement through BitLocker Drive Encryption configuration besides OS managed software encryption using AES-NI instruction sets.

NOTES

- Intel® TSE is currently a vPRO only feature and should be enabled on any NVMe attached to the platform PCIe port, including PCH ports.
- Intel® TSE works with PCIe NVMe storage devices off CPU and PCH Root Ports.
- Intel® TSE works with one PCIe NVMe storage device (regardless of Processor/ PCH).

5.16 Security Firmware Engines

5.16.1 Intel®Converged Security and Management Engine (Intel®CSME)

CSxE is security engine which provides security firmware authentication and loading, secure boot, platform debug control and manageability via Intel® Active Management Technology (Intel® AMT).

CSxE has a standalone small x86 processor, memory, crypto engine and I/O's.

CSxE is isolated in a secured hardware and firmware environment from the host processors.

5.16.2 Intel® Silicon Security Engine

A Security engine which is HW IP is based on CSxE HW IP and FW IP design to be silicon Root of Trust providing secure FW loading, measurements and on-tile certification authority.

The firmware is based on a new design which focus on security, simplicity of architecture and isolated environment.

5.16.3 Intel® Graphics System Controller (Intel® GSC)

Graphics System Controller (GSC) is a HW IP block embedded within the media IP block of the graphics component to support content and display protection services such as DRM and HDCP.

NOTE

All graphics security functionalities are handled by GSC which was previously implemented by CSxE.

5.16.4 Intel® Partner Security Engine

Intel® Partner Security Engine is a new security engine in which its' HW IP block is Intel® Silicon Security Engine based. Intel® Partner Security Engine IP is located on the processor and its purpose is to run or offload security-sensitive flows of the operating system.

6.0 Intel Virtualization Technology

Intel® Virtualization Technology (Intel® VT) makes a single system appear as multiple independent systems to software. This allows multiple, independent operating systems to run simultaneously on a single system. Intel® VT comprises technology components to support Virtualization of platforms based on Intel® architecture microprocessors and chipsets.

Intel® Virtualization Technology (Intel® VT) Intel® 64 and Intel® Architecture (Intel® VT-x) added hardware support in the processor to improve the Virtualization performance and robustness. Intel® Virtualization Technology for Directed I/O (Intel® VT-d) extends Intel® VT-x by adding hardware assisted support to improve I/O device Virtualization performance.

Intel® VT-x specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 3*. Available at:

http://www.intel.com/products/processor/manuals

The Intel® VT-d specification and other VT documents can be referenced at:

http://www.intel.com/content/www/us/en/virtualization/virtualization-technology/.

6.1 Intel® Virtualization Technology (Intel® VT) for Intel® 64 and Intel® Archit

Intel® VT-x Objectives

Intel® VT-x provides hardware acceleration for virtualization of IA platforms. Virtual Machine Monitor (VMM) can use Intel® VT-x features to provide an improved reliable virtualization platform. By using Intel® VT-x, a VMM is:

- **Robust:** VMMs no longer need to use para-virtualization or binary translation. This means that VMMs will be able to run off-the-shelf operating systems and applications without any special steps.
- **Enhanced:** Intel® VT enables VMMs to run 64-bit guest operating systems on IA x86 processors.
- **More Reliable:** Due to the hardware support, VMMs can now be smaller, less complex, and more efficient. This improves reliability and availability and reduces the potential for software conflicts.
- **More Secure:** The use of hardware transitions in the VMM strengthens the isolation of VMs and further prevents corruption of one VM from affecting others on the same system.

Intel® VT-x Key Features

The processor supports the following Intel ® VT-x features:

- **Mode-based Execute Control for EPT (MBEC)**

A mode of EPT operation which enables different controls for executability of Guest Physical Address (GPA) based on Guest specified mode (User/ Supervisor) of linear address translating to the GPA.

- **Extended Page Table (EPT) Accessed and Dirty Bits**

  EPT A/D bits enabled VMMs to efficiently implement memory management and page classification algorithms to optimize VM memory operations, such as defragmentation, paging, live migration, and check-pointing. Without hardware support for EPT A/D bits, VMMs may need to emulate A/D bits by marking EPT paging-structures as not-present or read-only, and incur the overhead of EPT page-fault VM exits and associated software processing.

- **EPTP (EPT pointer) switching**

  EPTP switching is a specific VM function. EPTP switching allows guest software (in VMX non-root operation, supported by EPT) to request a different EPT paging-structure hierarchy. This is a feature by which software in VMX nonroot operation can request a change of EPTP without a VM exit. The software will be able to choose among a set of potential EPTP values determined in advance by software in VMX root operation.

- **Pause loop exiting**

  Support VMM schedulers seeking to determine when a virtual processor of a multiprocessor virtual machine is not performing useful work. This situation may occur when not all virtual processors of the virtual machine are currently scheduled and when the virtual processor in question is in a loop involving the PAUSE instruction. The feature allows detection of such loops and is thus called PAUSE-loop exiting.

- **Extended Page Tables (EPT)**
  - EPT is hardware assisted page table virtualization.
  - It eliminates VM exits from guest OS to the VMM for shadow page-table maintenance.

- **Virtual Processor IDs (VPID)**
  - Ability to assign a VM ID to tag processor P/LP E core hardware structures (such as TLBs).
  - This avoids flushes on VM transitions to give a lower-cost VM transition time and an overall reduction in virtualization overhead.

- **Guest Preemption Timer**
  - The mechanism for a VMM to preempt the execution of a guest OS after an amount of time specified by the VMM. The VMM sets a timer value before entering a guest.
  - The feature aids VMM developers in flexibility and Quality of Service (QoS) guarantees.

- **Descriptor-Table Exiting**
  - Descriptor-table exiting allows a VMM to protect a guest OS from internal (malicious software based) attack by preventing the relocation of key system data structures like IDT (interrupt descriptor table), GDT (global descriptor table), LDT (local descriptor table), and TSS (task segment selector).
  - A VMM using this feature can intercept (by a VM exit) attempts to relocate these data structures and prevent them from being tampered by malicious software.

- **Hypervisor-Managed Linear Address Translation (HLAT)**
  - HLAT is active when the "enable HLAT" VM-execution control is 1. The processor looks up the HLAT if, during a guest linear address translation, the guest linear address matches the Protected Linear Range. The lookup from guest linear addresses to the guest physical address and attributes is determined by a set of HLAT paging structures.
  - The guest paging structure managed by the guest OS specifies the ordinary translation of a guest linear address to the guest physical address and attributes that the guest ring-0 software has programmed, whereas HLAT specifies the alternate translation of the guest linear address to guest physical address and attributes that the Secure Kernel and VMM seek to enforce. A logical processor uses HLAT to translate guest linear addresses only when those guest linear addresses are used to access memory (both for code fetch and data load/store) and the guest linear addresses match the PLR programmed by the VMM/Secure Kernel.
  - HLAT specifications and functional descriptions are included in the Intel® Architecture Instruction Set Extensions Programming Reference. Available at:

    https://software.intel.com/en-us/download/intel-architecture-instruction-set-extensions-programming-reference
- **Virtualization Exceptions**

  A virtualization exception is a new processor exception. It uses vector 20 and is abbreviated #VE. A virtualization exception can occur only in VMX non-root operation. Virtualization exceptions occur only with certain settings of certain VM-execution controls. Generally, these settings imply that certain conditions that would normally cause VM exits instead cause virtualization exceptions
- **Translation of Guest-Physical Addresses Used by Intel Processor Trace**

  With the "Intel PT uses guest physical addresses" feature , the addresses used by Intel PT can be treated as guest-physical addresses and translated using EPT. These addresses include the addresses of the output regions as well as the addresses of the ToPA entries that contain the output-region addresses.

6.2 Intel® Virtualization Technology (Intel® VT) for Directed IO (Intel® VT-d)

Intel® VT-d Objectives

The key Intel® VT-d objectives are domain-based isolation and hardware-based virtualization. A domain can be abstractly defined as an isolated environment in a platform to which a subset of host physical memory is allocated. Intel® VT-d provides accelerated I/O performance for a Virtualization platform and provides software with the following capabilities:

- **I/O Device Assignment and Security**: for flexibly assigning I/O devices to VMs and extending the protection and isolation properties of VMs for I/O operations.
- **DMA Remapping**: for supporting independent address translations for Direct Memory Accesses (DMA) from devices.
- **Interrupt Remapping**: for supporting isolation and routing of interrupts from devices and external interrupt controllers to appropriate VMs.
- **Reliability**: for recording and reporting to system software DMA and interrupt errors that may otherwise corrupt memory or impact VM isolation.

Intel® VT-d accomplishes address translation by associating transaction from a given I/O device to a translation table associated with the Guest to which the device is assigned. It does this by means of the data structure in the following illustration. This table creates an association between the device's PCI Express* Bus/Device/Function (B/D/F) number and the base address of a translation table. This data structure is populated by a VMM to map devices to translation tables in accordance with the device assignment restrictions above and to include a multi-level translation table (VT-d Table) that contains Guest specific address translations.

Figure 2. Device to Domain Mapping Structure in Legacy Mode

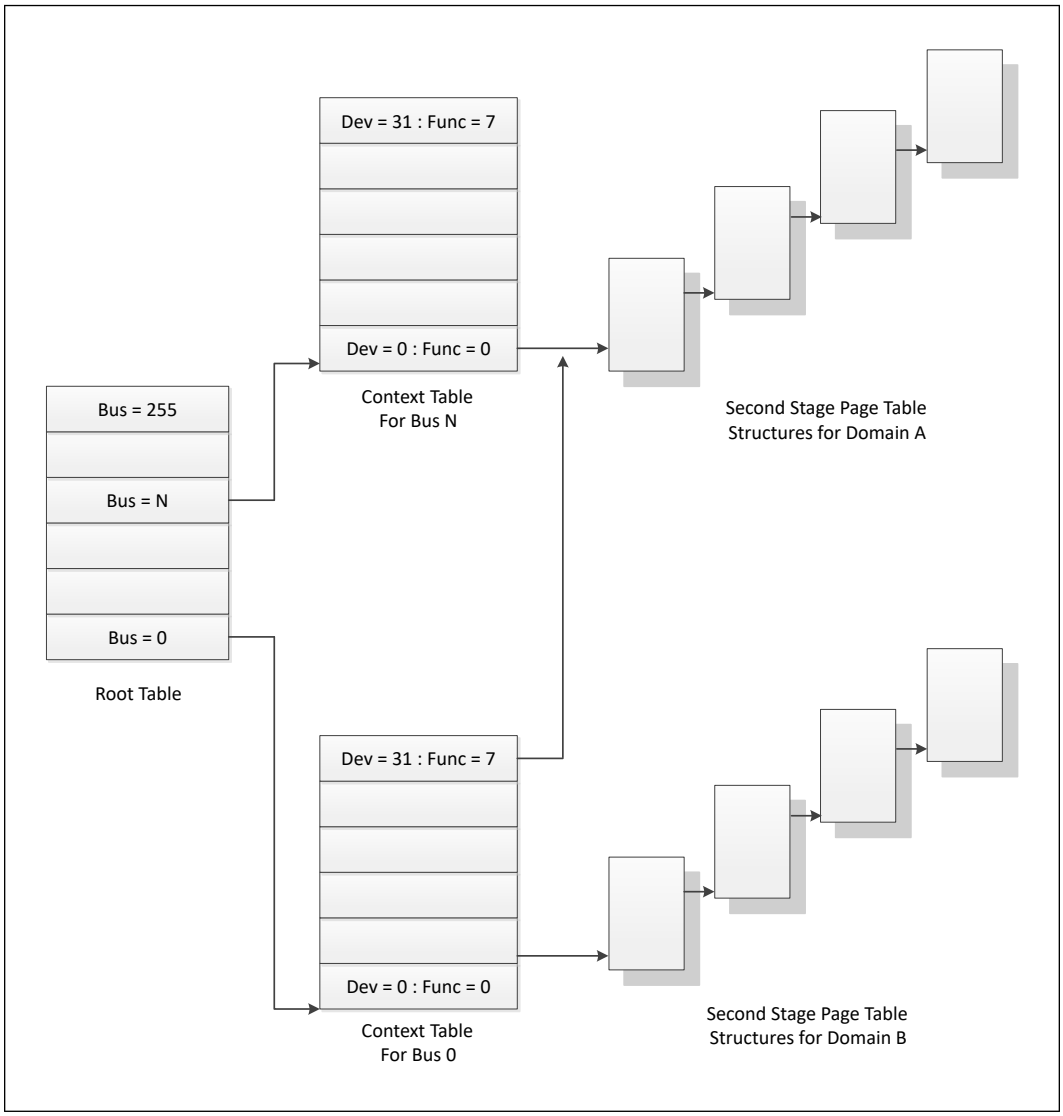
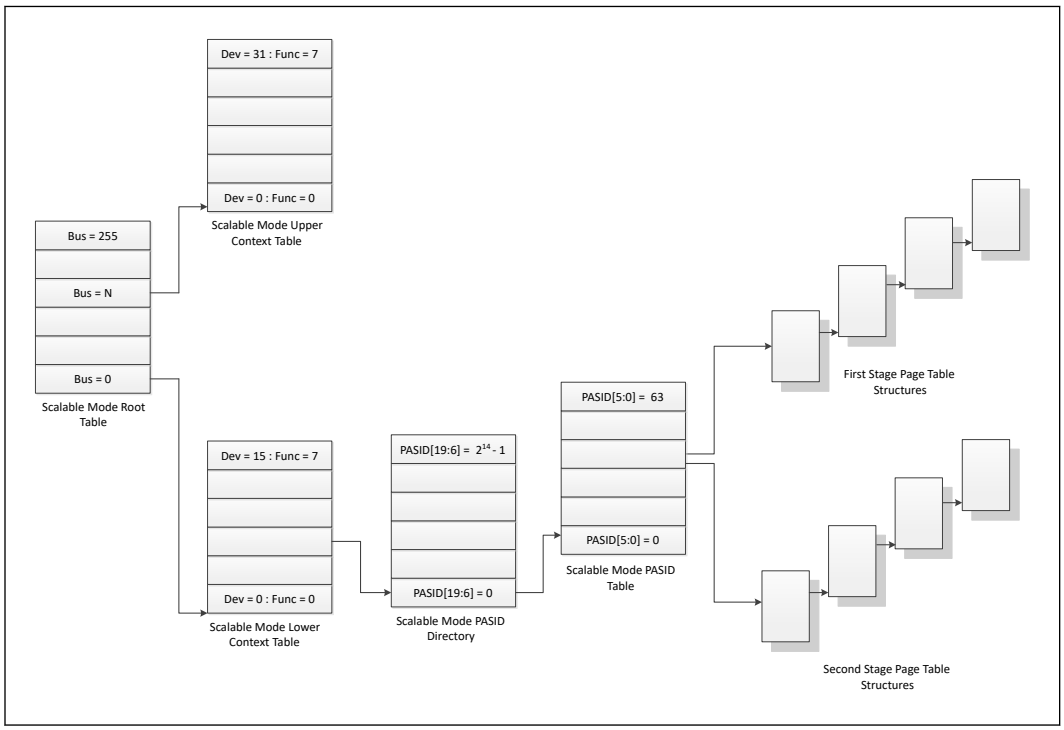


Figure 3. Device to Domain Mapping Structure in Scalable Mode



Intel® VT-d functionality often referred to as an Intel® VT-d Engine, has typically been implemented at or near a PCI Express* host bridge component of a computer system. This might be in a chipset component or in the PCI Express functionality of a processor with integrated I/O. When one such VT-d engine receives a PCI Express transaction from a PCI Express bus, it uses the B/D/F number associated with the transaction to search for an Intel® VT-d translation table. In doing so, it uses the B/D/F number to traverse the data structure shown in the above figure.

- If it finds a valid Intel® VT-d table in this data structure, it uses that table to translate the address provided on the PCI Express bus.
- If it does not find a valid translation table for a given translation, this results in an Intel® VT-d fault.

If Intel® VT-d translation is required, the Intel® VT-d engine performs an N-level table walk.

For more information, refer to Intel® Virtualization Technology for Directed I/O Architecture Specification:

http://www.intel.com/content/dam/www/public/us/en/documents/product-specifications/vt-directed-io-spec.pdf

Intel® VT-d Key Features

The processor supports the following Intel® VT-d features:

- Memory controller and processor graphics comply with the Intel® VT-d 4.0 Specification.
- 3 Intel® VT-d DMA Remapping Hardware Units:

  - — **GFx** DMA Remapping Hardware Unit: servicing iGD (Processor Graphics Dev2)
  - — **Non-GFX** DMA Remapping Hardware Unit: servicing NPU (Dev11), IPU (dev5), PMCS (dev4)
  - — Default DMA remap engine rest of PCI compatible devices and IOxAPIC + HPET
- 42-bit guest physical address and host physical address widths
- 4-level Intel® VT-d Page walk - all VTd engines support 4-level tables only (adjusted guest address width of 48 bits)
- Support Device-TLB for both engines for integrated accelerators (that is, GFX and Non-GFX)
- Support for register-based fault recording only (for single entry only) and support for MSI interrupts for faults
- Support for both leaf and non-leaf caching
- Support for non-caching of invalid page table entries
- Support for hardware-based flushing of translated but pending writes and pending reads, upon invalidation
- Support for all Queue based Invalidation descriptor types
- Support for Interrupt Remapping and Posted Interrupt
- Support Abort DMA Mode
- Support Performance Monitoring
- Intel® VT-d - All VTd engines support 4K, 2M and 1G page sizes
- Scalable Mode - All VTd engines support Scalable mode operation (using RID_PASID only)
- Nested - All Intel® VT-d engines support Nested translation

6.3 Intel® APIC Virtualization Technology (Intel® APICv)

APIC virtualization is a collection of features that can be used to support the virtualization of interrupts and the Advanced Programmable Interrupt Controller (APIC).

When APIC virtualization is enabled, the processor emulates many accesses to the APIC, tracks the state of the virtual APIC, and delivers virtual interrupts — all in VMX non-root operation without a VM exit.

The following are the VM-execution controls relevant to APIC virtualization and virtual interrupts:

- **Virtual-interrupt Delivery:** This controls enables the evaluation and delivery of pending virtual interrupts. It also enables the emulation of writes (memory-mapped or MSR-based, as enabled) to the APIC registers that control interrupt prioritization.
- **Use TPR Shadow:** This control enables emulation of accesses to the APIC's task-priority register (TPR) via CR8 and, if enabled, via the memory-mapped or MSR-based interfaces.

- **Virtualize APIC Accesses:** This control enables virtualization of memory-mapped accesses to the APIC by causing VM exits on accesses to a VMM-specified APIC-access page. Some of the other controls, if set, may cause some of these accesses to be emulated rather than causing VM exits.
- **Virtualize x2APIC Mode:** This control enables virtualization of MSR-based accesses to the APIC.
- **APIC-register Virtualization:** This control allows memory-mapped and MSR-based reads of most APIC registers (as enabled) by satisfying them from the virtual-APIC page. It directs memory-mapped writes to the APIC-access page to the virtual-APIC page, following them by VM exits for VMM emulation.
- **Process Posted Interrupts:** This control allows software to post virtual interrupts in a data structure and send a notification to another logical processor; upon receipt of the notification, the target processor will process the posted interrupts by copying them into the virtual-APIC page.

NOTE

Intel® APIC Virtualization Technology may not be available on all SKUs.

Intel® APIC Virtualization specifications and functional descriptions are included in the *Intel® 64 Architectures Software Developer's Manual, Volume 3*. Available at:

http://www.intel.com/products/processor/manuals

7.0 Instructions Set Enhancements

More information on Instruction Set Enhancements can be found at Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1:

https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

7.1 CMPccXADD

CMPccXADD is a new set of instructions that can be used to optimize certain highly contended synchronization scenarios that today use CMPXCHG (semaphores, shared queues, etc).

7.2 Linear Address Masking (LAM)

Linear Address Masking (LAM) repurposes the upper (untranslated) linear address bits to make them available for software use (for example, as metadata). This is accomplished by removing exiting canonical address checks when LAM is enabled.

When LAM not turned on, in 64-bit mode, linear address have 64 bits and are translated either with 4-level paging, which translates the low 48 bits of each linear address, or with 5-level paging, which translates 57 bits. The upper linear-address bits are reserved through the concept of canonicality. A linear address is 48-bit canonical if bits 63:47 of the address are identical; it is 57-bit canonical if bits 63:56 are identical. (Any linear address that is 48-bit canonical is also 57-bit canonical.)

When 4-level paging is active, the processor requires all linear addresses used to access memory to be 48-bit canonical; similarly, 5-level paging ensures that all linear addresses are 57-bit canonical.

Software usages that associate metadata with a pointer might benefit from being able to place metadata in the upper (untranslated) bits of the pointer itself. However, the canonicality enforcement mentioned earlier implies that software would have to mask the metadata bits in a pointer (making it canonical) before using it as a linear address to access memory or alternatively create Paging translation tables with redundancies.

LAM allows software to use pointers with metadata without having to mask the metadata bits. A LAM enabled processor when LAM is turned on by SW, internally ignores the metadata bits in a pointer before using it as a linear address to access memory. When LAM turned on by SW, the processor perform canonicality checks by only comparing bit 62 to bit 47 or to bit 56 (Depending on the paging mode) .

NOTE

LAM is supported only in 64-bit mode and applies only to addresses used for data accesses. LAM does not apply to addresses used for instruction fetches or to those that specify the targets of jump and call instructions.

7.3 SW Resource Prioritization

Resource Prioritization enables an OS to specify the priority of a thread to the processor P/LP E core. The processor P/LP E core may use the OS specified thread priority to optimize allocation of shared hardware resources based on priority. The processor decision on how to optimally allocate resources will be based on various system constraints in addition to the priority of other actively running threads. For example, a thread with higher priority may be allocated more of a resource when compared to a thread that has been tagged with a lower priority.

The following are key features of Resource Prioritization:

- Thread tagging interface to indicate priority of a thread
- Fixed 4 levels of priority
  - Priority 0 = Highest Priority
  - Priority 1 = Lower Priority than Priority 0
  - Priority 2 = Lower Priority than Priority 1
  - Priority 3 = Lowest Priority
- Priority Based Resource Allocation of shared hardware resources

8.0 Intel® Neural Processing Unit (Intel® NPU)

The NPU IP targets general Deep Learning inferencing applications in AI PC, connected devices and edge servers. It delivers the high processing throughput necessary to satisfy the demands of such applications.

The NPU technology is applicable to personal computing devices such as tablets, laptops and PCs as a way to introduce AI-based applications and services in power and performance sensitive platforms.

8.1 Functional Description

The NPU IP comprises several individual components grouped into two major subsystems:

1. Host Control
2. Deep Learning Accelerator

Details of these blocks are provided in the next sections.

Host Control

The functionality of the NPU is exposed to a SoC via a base set of registers (enumerated as a PCIe device or directly memory mapped into the address space of the Host).
These registers provide access to control and data path interfaces and reside in the Host Subsystem. All host communications are consumed by the NPU scheduler, a 64-bit RISC-V micro-controller. As well as responding to control messages it manages all the job submission/completion FIFOs that make up the data path of the NPU.

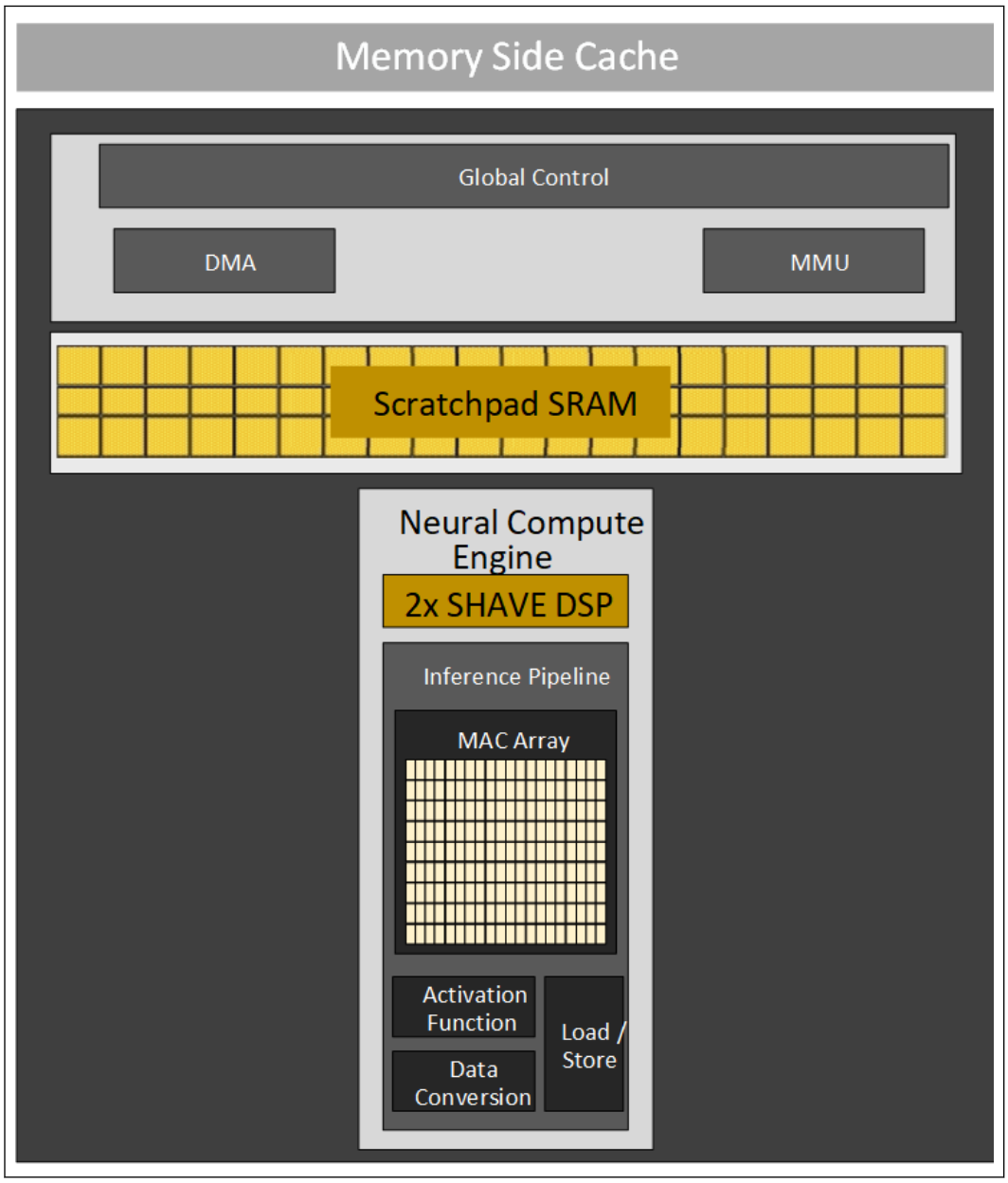
Deep Learning (Neural Compute Engine)

The NPU IP Deep Learning capability is provided by a configurable number of Neural Compute Engine (NCE) Tiles. The NCE Tiles are managed by the NPU Scheduler.

Each Tile includes a configurable amount of near-compute SRAM, one Data Processing Unit (DPU) with a configurable number of multiply-accumulates, and two DSPs (SHAVE-512) for optimal processing of custom deep learning operations. Global barriers and task FIFOs are also available for job synchronization and dispatch.
The Intel® Core™ Processor (Series 3) NPU comprises of 1 NCE Tiles, totaling 4k DPU INT8 MACs, 2 DSPs and 2.0 MB of associated near-compute memory.

Below is the block diagram of NPU IP:

Figure 4. NPU IP Block Diagram



8.1.1 HOST Control

The IP comprises several individual components such as CPU Subsystem, Host Subsystem and Host Interface.

The Host Subsystem is the main downstream interface between the NPU and the processor.

The Host Interface is the upstream interface between the NPU and the processor/ buttress.

The primary functions of the CPU subsystem are job scheduling and NPU resource management.

8.1.2 Deep Learning Accelerators (NCE)

The Neural Compute Engine (NCE) is a hardware accelerator for Deep Neural Network (DNN) workloads. It features a highly configurable pipeline for maximum support of DNN operations, such as Long Short-Term memory (LSTM) and Local Response Norm (LRN). It also leverages sparsity and low precision for optimal performance.

The Neural Compute Engine is built from one **Neural Compute Tile**, tile is a primary unit of computing.

For hardware assisted task synchronization, the NCE Subsystem provides barriers and workload FIFOs. Barriers remove as much software overhead as possible through ISR loops and programming sequences to keep the compute and data-movement pipelines full.

8.1.2.1 Feature Set

- 1 Neural Compute Engine Tile, where each tile consist of:
  - 2.0 MB CMX Memory
  - Two 512-bit ACT-SHAVEs
  - DPU (4K MACs)
- DMA Engine:
  - 2x 64B AXI Interface to DDR
  - Bit Compactor Compression unit for weights decompression and activation compression and decompression
  - Address Translation Prefetching capability in the DMA controller. A dedicated prefetch machine provides single-read accesses to pages at a configurable offset from the current transfer. All prefetch commands are carried on the read channel of the converged 64B AXI DMA data bus.
  - Address Patching capability for DRAM Accesses
- 256 KB of SHAVE L2 Cache for Data and Instruction shared between ACT-SHAVEs.
- Barriers for hardware and assisted task synchronization and pipelining
- Programmable HW FIFO Block for Work Descriptors and IPC
- Virtual Addressing for all resources used during Inference (Memory, Barriers, FIFOs and DMA/M2I Interrupt IDs)

8.1.2.2 NCE Tile

The NCE Tile is the building block of the NCE Subsystem. The Intel® Core™ Processor (Series 3) NCE subsystem supports one tile configuration. The NCE tile contains the following support:

- 2MB of NCE Tile Internal Storage.
- Single Data Processing Units (DPUs) where each DPU supports 4096 MACs.
- Two ACT-SHAVE DSPs with shared data and instruction L2 Cache (256 kB) used for flexible tensor compute operation.

8.1.2.3 ACTIVATION-SHAVE (ACT-SHAVE)

ACT-SHAVE is Intel Movidius DSP Processor which supports 512 bit vector operations. Two ACT-SHAVE DSPs are placed in each NCE Tile and are used for custom layer and standard layers that do not map well to the DPU.

8.1.2.4 Data Processing Unit (DPU)

The Data Processing Unit (DPU) supports 4k MACs built from 256 MAC Processing Engines (MPE) with 16 Fused-MACs in each MPE.

Feature Set

- SprLUT feature - SprLUT was added for complex non-linear activation functions.
- Optimized HW support for standard and depth-wise convolutions,
- Convolution Kernel size of M*N where N, M are up to 11,
- Convolution Stride of up to 8,
- Support configurable padding of activations,
- Supports reuse of activations and weights to reduce CMX Memory read bandwidth,
- Support for both Dense and Sparse operations,
- Sparse Element-wise operations,
- Supported Data Types
  - FP8
  - I8
  - U8
- Precision
  - FP32 and INT32 accumulators
  - Floating point and integer scaling supported
  - Integer floating point inline conversion
  - Floating point subnormal support
- Supports statistics gathering (Hardware Profiling)
- Features 64 Post Processing Elements (PPE) where each support:
  - Quantization
  - Activation function
  - Element-Wise functions
- Features Four 256-bit wide Read-Only Ports and four 256-bit wide Read/Write Ports
- Support for Sparse acceleration and compression to increase effective TOPs by up to 2x.
  - Sparsity awareness allows the MAC circuits to run more TOPs by not consuming cycles processing data that does not affect the result.
  - Those extra (or effective) TOPs translate to lower power for the same compute performance, or, higher compute performance for the same power by comparison to a design that is sparsity agnostic.

- DPU State Machine is responsible for loading tensor workloads and micro-scheduling.
- New Feature in NPU that allows splitting Input channels (ICs) across MPE to improve the overall utilization of the PE array.

9.0 Audio Voice and Speech

The AVS subsystem builds upon the AVS features of previous platforms to provide a richer user experience. This section will cover the HW features used in the Processor for use within the AVS subsystem. The AVS subsystem consists of a collection of controller, DSP, memory, and link interfaces that provides the audio experience to the platform. This subsystem provides streaming of audio from the host SW to external audio codecs with the host processor and/or DSP providing the audio enrichment.

The optional DSP can be enabled in the audio subsystem to provide low latency HW/FW acceleration for common audio and voice functions such as audio encode/decode, acoustic echo cancellation, noise cancellation, etc. With such acceleration, the integration of the AVS subsystem into the processor is expected to provide longer music playback times and VOIP call times for the platform.

The key HW features of the AVS Subsystem are described in the following topics:

- Intel® High Definition Audio (Intel® HD Audio) Controller Capabilities
- Audio DSP Capabilities
- Intel® High Definition Audio Interface Capabilities
- USB Audio Offload Support
- Intel® Display Audio Interface
- MIPI* SoundWire* Interface

Table 16. Acronyms

Acronyms	Description
DMA	Direct Memory Access.
DMIC	Digital Microphone. PDM based MEMs microphone modules.
DSP	Digital Signal Processor. In AVS specifically a DSP to process audio data.
MEMs	Micro electrical mechanical Systems. For AVS devices such as Digital MEMs Microphones.
MSI	Message Signaled Interrupt. An in-band method of signaling an interrupt.
PCM	Pulse Code Modulation. Modulation with amplitude coded into stream.
PDM	Pulse Density Modulation. Modulation with amplitude coded by pulse density.
SDI	Serial Data In.
SDO	Serial Data Out.
VOIP	Voice Over Internet Protocol

Table 17. References

Specification	Location
Intel® High Definition Audio Specification	http://www.intel.com/content/www/us/en/standards/high-definition-audio-specification.html

9.1 Intel® High Definition Audio (Intel® HD Audio) Controller Capabilities

The Intel® HD Audio controller is the standard audio host controller widely adopted in the PC platform, with industrial standard Intel® HD Audio driver software available for Microsoft* Windows* and many other Linux* based Operating Systems. With the converged audio architecture initiatives, it is also the baseline audio host controller for phone and tablet platforms with optional DSP support. Intel® HD Audio controller capabilities are listed as follows:

- PCI / PCI Express* controller
  - Option to hide PCI configuration space an use ACPI method for enumeration
- Supports data transfers, descriptor fetches, and DMA position writes using VC0 or VC1
- Independent Bus Host logic for 20 general purpose DMA streams: 11 input and 9 output
- Supports variable length stream slots
- Each general purpose stream supports up to:
  - 16 channels per stream
  - 32 bits/sample
  - 192 kHz sample rate
- Supports memory-based command/response transport
- Supports optional Immediate Command/Response mechanism
- Supports input and output stream synchronization
- Supports MSI interrupt delivery
- Support for ACPI D3 and D0 Device States
- Supports Function Level Reset (FLR)
  - Only if exposed as a PCI Express device (or ACPI method)
- Support Converged Platform Power Management (CPPM)
  - Support 1 ms of buffering with all DMA running with maximum bandwidth.

9.2 Audio DSP Capabilities

The Audio DSP offload engine is a feature providing low power DSP functionality and offloads the audio processing operation from the host CPU. It is exposed as an optional capability feature under the Intel® HD Audio controller, allowing the enumeration through the Intel® HD Audio driver software (if implemented). Audio DSP capabilities are listed as follows:

- Up to 3 x 614 MHz Tensilica* LX7 HIFI4 DSP Cores
- Up to 3.46 MB of L2 HP SRAM for all DSP Cores
- L2 uncache memory accessing up to 16 x 16 MB of remote DDR region
- DSP offload for low power audio rendering and recording
- Various DSP functions provided by Tensillica Core: MP3, AAC, Dolby Digital*, etc.
- Host downloadable DSP FW functions

- Voice call processing enhancement
- HW based DSP accelerators, for example, Machine Learning block and SHA engine

9.3 Intel® High Definition Audio Interface Capabilities

The Intel® HD Audio interface is a feature offering connections to the compatible codecs. The Intel® HD Audio compatible codecs are widely available from various vendors allowing PC platform OEM's to choose them based on features, power, cost consideration. The audio codec can work with the in-box Intel® HD Audio driver software provided in various Operating Systems providing a seamless user experience. These Intel® HD Audio compatible codecs will be enumerated by the Intel® HD Audio driver software (if discovered over the Intel® HD Audio interface). Intel® HD Audio interface capabilities are listed as follows:

- The SDI signals to support external codecs.
- Drives variable frequency (6 MHz to 24 MHz) BCLK to support:
  - SDO double pumped up to 48 Mb/s
  - SDIs single pumped up to 24 Mb/s
- Provides cadence for 44.1 kHz-based sample rate output
- Supports LV Mode (1.8 V)

9.4 Direct Attached Digital Microphone (PDM) Interface

The direct attached digital microphone interface is a feature offering connections to PDM based digital microphone modules without the need of audio codecs. This provides the lowest possible platform power with the decimation functionality integrated into the audio host controller. Features for the digital microphone interface are listed as follows:

- Up to 2 Digital Mic Ports with to 2 Digital Mic Modules per Digital Mic Port, Audio processing DSP and SRAM support is limited to 2 Digital Mic.
- Ability to combine multiple Digital Mic Ports to for mic arrays that are synchronized on sampling rate basis
- 2 PCM Audio Streams with independent PCM sampling rates per Digital Mic Port
- Ability to map each Digital Mic Port stereo PCM streams output to a sub-set of a multi-channel PCM stream data transferred to an Audio Link Hub
  - Support dynamic scaling up/down of microphone channels array after the stream has started
- Support child clock input mode of operation

9.5 USB Audio Offload Support

USB Audio Offload provides audio mixing / processing support for USB audio endpoint connected through the xHCI Controller. This is aimed at providing a universal audio offload power benefit across various audio devices connected to the platform and USB audio usage is expected to gain more popularity with the introduction of USB Type-C* connector. These USB audio endpoint will be enumerated by the xHCI Controller SW and only the audio streaming path is peer to the Audio DSP subsystem for DSP FW mixing / processing support. USB Audio Offload capabilities are listed as follows:

- Up to 2 audio output streams support
- Up to 4 audio input streams support
- Provides cadence for 44.1 kHz-based sample rate output
- Support isochronous audio stream offload for LS / FS / HS USB audio device
- Support synchronous / asynchronous / adaptive modes of isochronous audio streaming
- Support non-PCM encoded audio bit stream defined by IEC61937 / IEC60958 standard
  - Packetizing into PCM sample format and PCM equivalent rates

9.6 I2S PCM Interface

The I^2S / PCM interface is a feature offering connection to the I^2S / PCM audio codecs. The I^2S / PCM audio codecs are widely adopted in the phone and tablet platforms as they are typically customized for low power application. The codec structure is typically unique per codec vendor implementation and requires vendor specific SW module for controlling the codec. These I^2S / PCM audio codecs will be enumerated based on ACPI table or OS specific static configuration information. The Audio DSP is required to be enabled in order to enable. I^2S / PCM Interface capabilities are listed as follows:

- Up to 3 bi-directional I^2S / PCM ports to support up to 16 channels per port
- Controller/device mode support for run-time selection
- Each I^2S / PCM ports are able to support multiple devices using PCM mode (also known as TDM Mode)
- Support multi I^2S / PCM port synchronization

9.7 Intel® Display Audio Interface

The Intel® iDisp Audio link is a feature offering connection to the Intel® iDisp Audio codec. The Intel® iDisp Audio codec is used to provide audio streams routing to the integrated HDMI and DP links, through the existing Intel® HD Audio controller SW stacks. This iDisp audio codec used to be attached to the Intel® HD Audio link, however, it transitioned to a dedicated 3-wire iDisp Audio link to save pin counts on the compute tile, as well as providing finer grain power management to the audio link interfaces. The Intel® iDisp Audio codec is enumerated by the Intel® HD Audio driver software. Features for the Intel® HD Audio interface is provided below:

- 1 SDI signal to support 1 iDisp audio codec
- Drives variable frequency (6 MHz to 96 MHz) BCLK to support
  - SDO single pumped to 96 Mb/s
  - SDI single pumped up to 96 Mb/s
- Provides cadence for 44.1 kHz-based sample rate output.

9.8 MIPI® SoundWire Interface

The SoundWire interface is a feature offering connection to the SoundWire devices, which include audio codecs and modem codecs. The SoundWire interface is the latest audio interface targeting (but not limited to) the phone and tablet market and the

main advantage is the connection simplicity with a two wires multi-drop topology and PDM streaming capabilities. There is also an option to increase the bandwidth by implementing additional data lane wires, up to 4 data lanes per SoundWire interface. SoundWire device class initiative for audio is bringing standardization to the audio codec SW stack. These devices are enumerated based on vendor / device ID of the SoundWire device reporting, allowing vendor customization of audio codec SW if desired. SoundWire interface capabilities are listed as follows:

- Up to 4 SoundWire interfaces frame rate synchronized on global periodic events
- Support wide range of bus frequency. Maximum frequency supported is 12.288 MHz.
- SNDW3 supports 4 data lane per SoundWire interface and SNDW2 supports 3 data lane per SoundWire interface
- Integrated SoundWire for iDiSP Audio functionality.
- Support SoundWire Device Class Specification for Audio Controls and Memories
- Up to 15 PCM bidirectional streams per SoundWire interface
  - — Direction is programmable as either input or output stream
- Up to 8 channels per PCM streams
- Interrupt / PME wake capable on DATA pin assertion in low power state

9.9 Signal Description

Signal Name	Type	Description
Intel® High Definition Audio Signals		
GPP_D16/**HDA_RST#**/DMIC_CLK_A1	O	**Intel HD Audio Reset**: Host H/W reset to internal and external codecs.
GPP_D11/**HDA_SYNC**/I2S0_SFRM	O	**Intel HD Audio Sync**: 48 kHz fixed rate frame sync to the codecs.
GPP_D10/**HDA_BCLK**/I2S0_SCLK	O	**Intel HD Audio Bit Clock**: Up to 24 MHz serial data clock generated by the Intel® HD Audio controller.
GPP_D12/**HDA_SDO**/I2S0_TXD	O	**Intel HD Audio Serial Data Out**: Serial TDM data output to the codecs. The serial output is double-pumped for a bit rate of up to 48 Mb/s.
GPP_D13/**HDA_SDI0**/I2S0_RXD	I/O	**Intel HD Audio Serial Data In 0**: Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated Pull-down resistors, which are enabled while the primary well is powered.
GPP_D17/**HDA_SDI1**/DMIC_DATA1	I/O	**Intel HD Audio Serial Data In 1**: Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated Pull-down resistors, which are enabled while the primary well is powered.
I²S / PCM Interface		
GPP_D10/HDA_BCLK/**I2S0_SCLK**	I/O	**I²S / PCM serial bit clock 0**: Serial bit clock used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).

continued...

Signal Name	Type	Description
GPP_S02/SNDW3_DATA1/SNDW0_CLK/ DMIC_CLK_A0/**I2S1_SCLK**	I/O	**I²S / PCM serial bit clock 1**:Serial bit clock is used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_S04/SNDW2_CLK/DMIC_CLK_A0/ **I2S2_SCLK**	I/O	**I²S / PCM serial bit clock 2**: Serial bit clock is used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D11/HDA_SYNC/**I2S0_SFRM**	I/O	**I²S / PCM serial frame indicator 0**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_S03/SNDW3_DATA2/ SNDW2A_DATA1/SNDW0_DATA0/ DMIC_DATA0/**I2S1_SFRM**	I/O	**I²S / PCM serial frame indicator 1**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_S05/SNDW2_DATA0/DMIC_DATA0/ **I2S2_SFRM**	I/O	**I²S / PCM serial frame indicator 2**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D12/HDA_SDO/**I2S0_TXD**	O	**I²S / PCM transmit data (serial data out)0**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_S00/SNDW3_CLK/**I2S1_TXD**	O	**I²S / PCM transmit data (serial data out)1**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_S06/SNDW2_DATA1/SNDW1_CLK/ DMIC_CLK_A1/**I2S2_TXD**	O	**I²S / PCM transmit data (serial data out)2**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_D13/HDA_SDI0/**I2S0_RXD**	I	**I²S / PCM receive data (serial data in)0**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_S01/SNDW3_DATA0/**I2S1_RXD**	I	**I²S / PCM receive data (serial data in)1**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_S07/SNDW3_DATA3/ SNDW2_DATA2/SNDW1_DATA0/ DMIC_DATA1/**I2S2_RXD**	I	**I²S / PCM receive data (serial data in)2**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_D09/**I2S_MCLK1_OUT**	O	**I²S / PCM Host reference clock 0**: This signal is the host reference clock that connects to an audio codec.
DMIC Interface		
GPP_S02/SNDW3_DATA1/SNDW0_CLK/ **DMIC_CLK_A0**/I2S1_SCLK or GPP_S04/SNDW2_CLK/**DMIC_CLK_A0**/ I2S2_SCLK	O	**Digital Mic Clock A0**:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz. Duplication for clock pin (instance A) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance A) should be used.
GPP_D16/HDA_RST#/**DMIC_CLK_A1** or GPP_S06/SNDW2_DATA1 /SNDW1_CLK/ **DMIC_CLK_A1**/I2S2_TXD	O	**Digital Mic Clock A1**:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz.

continued...

Signal Name	Type	Description
		Duplication for clock pin (instance A) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance A) should be used.
GPP_S03/SNDW3_DATA2/ SNDW2A_DATA1/SNDW0_DATA0/ **DMIC_DATA0**/I2S1_SFRM or GPP_S05/SNDW2_DATA0/ **DMIC_DATA0**/I2S2_SFRM	I	**Digital Mic Data**:Serial data input from the digital mic.
GPP_D17/HDA_SDI1/**DMIC_DATA1** or GPP_S07/SNDW3_DATA3/ SNDW2_DATA2/SNDW1_DATA0/ **DMIC_DATA1**/I2S2_RXD	I	**Digital Mic Data**:Serial data input from the digital mic.
Mic Mute Interface		
GPP_H03/**MIC_MUTE**		**Mic Mute**: Indicate the user privacy mode setting on the system
GPP_H17/**MIC_MUTE_LED**		**Mic Mute Led**: Led to Indicate the user privacy mode setting in the system.
SoundWire Interface		
GPP_S02/**SNDW3_DATA1**/ **SNDW0_CLK**/DMIC_CLK_A0/I2S1_SCLK	I/O	**SoundWire0 Clock**: Serial bit clock used to control the timing of a transfer. **SoundWire3 Multilane Data1**: To support multilane capability for high fidelity codecs
GPP_S03/**SNDW3_DATA2**/ **SNDW2A_DATA1**/**SNDW0_DATA0**/ DMIC_DATA0/I2S1_SFRM	I/O	**SoundWire3 Multilane Data2**: To support multilane capability for high fidelity codecs **SoundWire2 Multilane Data1**: To support multilane capability for high fidelity codecs **SoundWire0 Data0**: Serialized data line containing framing and data being transmitted/received.
GPP_S06/**SNDW2_DATA1** / **SNDW1_CLK**/DMIC_CLK_A1/I2S2_TXD	I/O	**SoundWire2 Multilane Data1**: To support multilane capability for high fidelity codecs **SoundWire1 Clock**: Serial bit clock used to control the timing of a transfer.
GPP_S07/**SNDW3_DATA3**/ **SNDW2_DATA2**/**SNDW1_DATA0**/ DMIC_DATA1/I2S2_RXD	I/O	**SoundWire3 Multilane Data3**: To support multilane capability for high fidelity codecs **SoundWire2 Multilane Data1**: To support multilane capability for high fidelity codecs **SoundWire1 Data0**: Serialized data line containing framing and data being transmitted/received.
GPP_S04/**SNDW2_CLK**/DMIC_CLK_A0/ I2S2_SCLK	I/O	**SoundWire2 Clock**: Serial bit clock used to control the timing of a transfer.
GPP_S05/**SNDW2_DATA0**/ DMIC_DATA0/I2S2_SFRM	I/O	**SoundWire2 Data0**: Serialized data line containing framing and data being transmitted / received.
GPP_S00/**SNDW3_CLK**/I2S1_TXD	I/O	**SoundWire3 Clock**: Serial bit clock used to control the timing of a transfer.
GPP_S01/**SNDW3_DATA0**/I2S1_RXD	I/O	**SoundWire3 Data0**: Serialized data line containing framing and data being transmitted / received.
SNDW_RCOMP	A	SoundWire Resistor compensation.

9.10 Integrated Pull-Ups and Pull-Downs

Table 18. Integrated Pull-Ups and Pull-Downs

Signal Name	Resistor Type	Value
HDA_SYNC	Pull-down	20 kohm
HDA_SDO	Pull-down	20 kohm
HDA_SDI[1:0]	Pull-down	20 kohm
DMIC_DATA[1:0]	Pull-down	20 kohm
SNDW[1:0]_DATA0	Pull-down	5 kohm
SNDW2_DATA[2:0]	Pull-down	5 kohm
SNDW3_DATA[3:0]	Pull-down	5 kohm

9.11 IO Signal Planes and States

Table 19. I/O Signal Planes and States

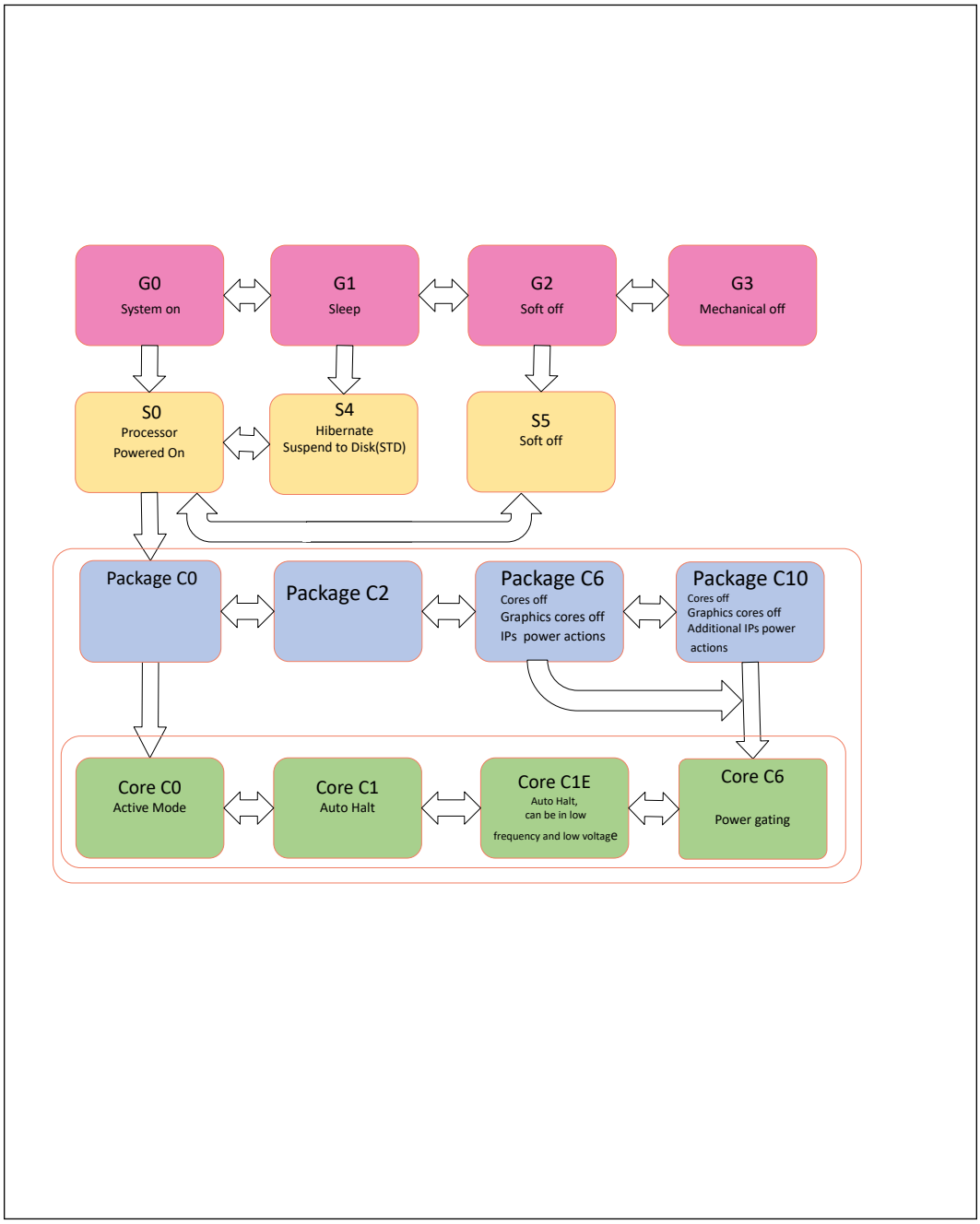
Signal Name	Power Plane	During Reset[1]	Immediately After Reset[1]	S4/S5
High Definition Audio Interface				
HDA_RST#	Primary	Asserted	Asserted	Asserted
HDA_SYNC	Primary	Internal Pull-down	Driven Low	Internal Pull-down
HDA_BCLK	Primary	Driven Low	Driven Low	Driven Low
HDA_SDO	Primary	Internal Pull-down	Driven Low	Internal Pull-down
HDA_SDI[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
DMIC Interface				
DMIC_CLK_A[1:0]	Primary	Driven Low	Driven Low	Driven Low
DMIC_DATA[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SoundWire Interface				
SNDW[1:0]_DATA0	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SNDW2_DATA[2:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SNDW3_DATA[3:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SNDW[3:0]_CLK	Primary	Driven Low	Driven Low	Driven Low
MIC_MUTE	Primary	Driven Low	Driven Low	Driven Low
MIC_MUTE_LED	Primary	Driven Low	Driven Low	Driven Low
Note: Reset reference for primary well pins is RSMRST#.				

10.0 Power Management

Table 20. References

Specification	Location
Advanced Configuration and Power Interface (ACPI)	https://uefi.org/sites/default/files/resources/ACPI_Spec_6_5_Aug29.pdf

Figure 5. Power State Block Diagram



10.1 System Power States, Advanced Configuration and Power Interface (ACPI)

This section describes System Power States and ACPI states supported by the processor.

Table 21. General System Power States

State	Description
G0/S0/C0	**Full On**: CPU operating. Individual devices may be shut to save power. The different CPU operating levels are defined by Cx states.
G0/S0/Cx	**Cx state**: CPU manages C-states by itself and can be in low power state
G0/S0ix/Cx	**S0ix**: The south supports an S0ix state which also requires the CPU be in a Cx state. Additional south power actions such as voltage reduction.
G1/S4	**Suspend-To-Disk (STD)**: The context of the system is maintained on the disk. All power is then shut to the system except to the logic required to resume. Externally appears same as S5 but may have different wake events.
G2/S5	**Soft Off**: System context not maintained. All power is shut except for the logic required to restart. A full boot is required when waking.
G3	**Mechanical OFF**: System context not maintained. All power shut except for the RTC. No "Wake" events are possible because the system does not have any power. This state occurs if the user removes the batteries, turns off a mechanical switch, or if the system power supply is at a level that is insufficient to power the "waking" logic. When system power returns the transition will depend on the state just prior to the entry to G3.

The table below shows the transitions rules among the various states.

NOTE

Transitions among the various states may appear to temporarily transition through intermediate states. For example, in going from S0 to S5, it may appear to pass through the G1/S4 state. These intermediate transitions and states are not listed in the table below.

Table 22. State Transition Rules for the Processor

Present State	Transition Trigger	Next State
G0/S0/C0	• SLP_EN bit set • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/Cx • G1/S4, or G2/S5 state • G2/S5 • G3
G0/S0/Cx	• Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0 • S5 • G3
G0/S0ix/Cx	• Any south action which is blocked from occurring while in S0ix [3] • CPU or south IP request for CPU C-state exit	• G0/S0/Cx • G0/S0/C2(or C0)
G1/S4	• Power Button Override[3] • Mechanical Off/Power Failure	• G2/S5 • G3
G2/S5	• Any Enabled Wake Event	• G0/S0/C0[2]

continued...

Present State	Transition Trigger	Next State
	• Mechanical Off/Power Failure	• G3
G3	• Power Returns	• S0/C0 (reboot) or G2/S5[4] (stay off until power button pressed or other wake event)[1,2]

Notes: 1. Some wake events can be preserved through power failure.
2. Transitions from the S4-S5 states to the S0 state are deferred until BATLOW# is inactive.
3. Includes all other applicable types of events that force the host into and stay in G2/S5.
4. If the system was in G1/S4 before G3 entry, then the system will go to S0/C0 or G1/S4.
5. On G3 exit, prior to the first transition to S0, S5 power may be higher than S5 power after the first S0 to S5 transition.
Some processor settings required to achieve minimum S5 power are loaded during first boot to S0 after a G3 exit. Consequently, entry into S5 from S0 will result in a more power-optimized S5 state than entry into S5 from G3 without an S5-S0-S5 transition. The difference is expected to be in the few mW range

System Power Planes

The system has several independent power planes, as described in the table below.

NOTE
When a particular power plane is shut off, it should go to a 0 V level.

Table 23. System Power Plane

Plane	Controlled By	Description
Memory	SLP_S4# signal SLP_S5# signal	When SLP_S4# goes active, power can be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory can also be shut down. When SLP_S5# goes active, power can be shut off to any circuit not required to wake the system from the S5 state. Since the memory context does not need to be preserved in the S5 state, the power to the memory can also be shut down.
Intel® CSME	SLP_A#	SLP_A# signal is asserted when the Intel® CSME goes to M-Off or M3-PG. Depending on the platform, this pin may be used to control power to various devices that are part of the Intel® CSME sub-system in the platform.
DEVICE[n]	GPIO	Individual subsystems may have their own power plane. For example, GPIO signals may be used to control the power to disk drives, audio amplifiers, or the display screen.

10.2 Functional Description

10.2.1 Features

- Support for *Advanced Configuration and Power Interface (ACPI)* providing power and thermal management
- PCI PME# signal for Wake Up from Low-Power states
- System Sleep State Control
  - — ACPI S4 state – Suspend-to-Disk (STD)
  - — ACPI G2/S5 state – Soft Off (SOFF)

  - — Power Failure Detection and Recovery
- Intel® CSME Power Management Support
  - — Wake events from the Intel® CSME (enabled from all S-States including Catastrophic S5 conditions)

10.2.2 Power Saving Features

Power Management Substates

A set of new features define new S0ix substates that provide lower power at a higher exit latency cost and, in some cases, fewer allowed wake events. The substates are denoted by suffixes appended to the S0i2 base name. The highest suffix number indicates the deepest substate. On the Intel® Core™ Processor (Series 3), the supported suffixes are S0i2.0, S0i2.1, S0i2.2. During the transition between S0 and Sx, the S0ix Substates logic is reconfigured to work in Sx.

S0ix in Sx

All the power saving features of S0ix are activated in Sx as well.

Naming Convention

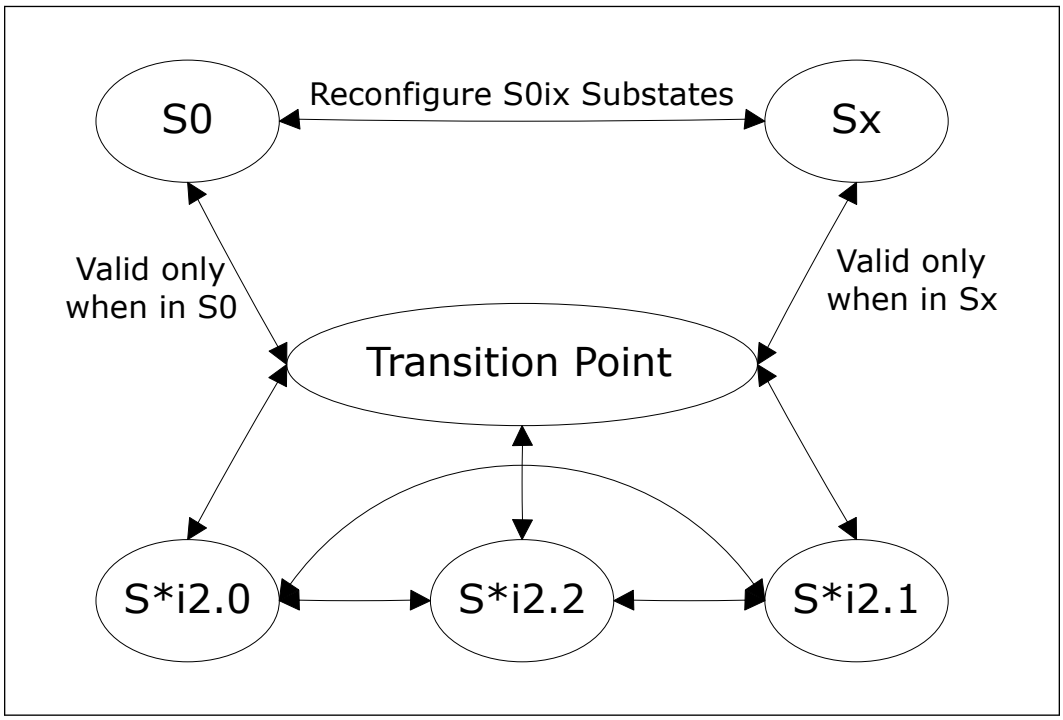
The naming convention: S*ix.y refers to any combination of S0/Sx and Substate.

Specifically:

- * represents any S0-Sx state (for example: S0, S4, S5)
- x represents any S0ix State (for example: S0i2)
- y represents any Substate (for example: .0, .1, .2,)

For example, to represent the "2.0" equivalent substate in any S0 or Sx state, use the naming S*i2.0

Figure 6. Power Management Substates



10.2.3 SMI SCI Generation

Upon any enabled SMI event taking place while the End of SMI (EOS) bit is set, the processor will clear the EOS bit and assert SMI , which will cause it to enter SMM space. SMI assertion is performed using a Virtual Legacy Wire (VLW) message.

Once the SMI VLW has been delivered, the processor takes no action on behalf of active SMI events until Host software sets the End of SMI (EOS) bit. At that point, if any SMI events are still active, the processor will send another SMI VLW message.

The SCI is a level-mode interrupt that is typically handled by an ACPI-aware operating system. In non-APIC systems (which is the default), the SCI IRQ is routed to one of the 8259 interrupts (IRQ 9, 10, or 11). The 8259 interrupt controller must be programmed to level mode for that interrupt.

In systems using the APIC, the SCI can be routed to interrupts 9, 10, 11, 20, 21, 22, or 23. The interrupt polarity changes depending on whether it is on an interrupt shareable with a PIRQ or not. The interrupt remains asserted until all SCI sources are removed.

The table below shows which events can cause an SMI and SCI.

NOTE

Some events can be programmed to cause either an SMI or SCI. The usage of the event for SCI (instead of SMI) is typically associated with an ACPI-based system. Each SMI or SCI source has a corresponding enable and status bit.

Table 24. Causes of SMI and SCI

Cause	SCI	SMI	Additional Enables[1]	Where Reported
PME#	Yes	Yes	PME_EN=1	PME_STS
PME_B0 (Internal, Bus 0, PME-Capable Agents)	Yes	Yes	PME_B0_EN=1	PME_B0_STS
PCI Express* PME Messages	Yes	Yes	PCI_EXP_EN=1 (Not enabled for SMI)	PCI_EXP_STS
PCI Express* Hot-Plug Message	Yes	Yes	HOT_PLUG_EN=1 (Not enabled for SMI)	HOT_PLUG_STS
Power Button Press	Yes	Yes	PWRBTN_EN=1	PWRBTN_STS
Power Button Override [6]	Yes	No	None	PWRBTNOR_STS
RTC Alarm	Yes	Yes	RTC_EN=1	RTC_STS
ACPI Timer overflow (2.34 seconds)	Yes	Yes	TMROF_EN=1	TMROF_STS
GPIO	Yes	Yes	Refer to Note 8	
LAN_WAKE#	Yes	Yes	SCI_EN=0, LAN_WAKE_EN=1	LAN_WAKE_STS
TCO SCI message from processor	Yes	No	None	CPUSCI_STS
TCO SCI Logic	Yes	No	TCOSCI_EN=1	TCOSCI_STS
TCO SMI Logic	No	Yes	TCO_EN=1	TCO_STS
TCO SMI – Year 2000 Rollover	No	Yes	None	NEWCENTURY_STS
TCO SMI – TCO TIMEROUT	No	Yes	None	TIMEOUT
TCO SMI – OS writes to TCO_DAT_IN register	No	Yes	None	OS_TCO_SMI
TCO SMI – NMI occurred (and NMIs mapped to SMI)	No	Yes	NMI2SMI_EN=1	TCO_STS, NMI2SMI_STS
TCO SMI – INTRUDER# signal goes active	No	Yes	INTRD_SEL=10	INTRD_DET
TCO SMI – Changes of the WPD (Write Protect Disable) bit from 0 to 1	No	Yes	LE (Lock Enable)=1	BIOSWR_STS
TCO SMI – Write attempted to BIOS	No	Yes	WPD=0	BIOSWR_STS
BIOS_RLS written to 1 [7]	Yes	No	GBL_EN=1	GBL_STS
GBL_RLS written to	No	Yes	BIOS_EN=1	BIOS_STS
Write to B2h register	No	Yes	APMC_EN = 1	APM_STS
Periodic timer expires	No	Yes	PERIODIC_EN=1	PERIODIC_STS
64 ms timer expires	No	Yes	SWSMI_TMR_EN=1	SWSMI_TMR_STS
Enhanced USB Legacy Support Event	No	Yes	LEGACY_USB2_EN = 1	LEGACY_USB2_STS
Serial IRQ SMI reported	No	Yes	None	SERIRQ_SMI_STS
Device monitors match address in its range	No	Yes	Refer to DEVTRAP_STS register description	DEVTRAP_STS
SMBus Host Controller	No	Yes	SMB_SMI_EN, Host Controller Enabled	SMBus host status reg.
SMBus SMBALERT# signal active	No	Yes	None	SMBUS_SMI_STS

continued...

Cause	SCI	SMI	Additional Enables[1]	Where Reported
BATLOW# assertion	Yes	Yes	BATLOW_EN=1	BATLOW_STS
Access microcontroller 62h/66h	No	Yes	MCSMI_EN	MCSMI_STS
SLP_EN bit written to 1	No	Yes	SMI_ON_SLP_EN=1	SMI_ON_SLP_EN_STS
SPI Command Completed	No	Yes	None	SPI_SMI_STS
eSPI SCI/SMI Request [9]	Yes	Yes	eSPI_SCI_EN	eSPI_SCI_STS eSPI_SMI_STS
Software Generated GPE	Yes	Yes	SWGPE_EN=1	SWGPE_STS
Intel® CSME	Yes	Yes	CSME_SCI_EN=1 CSME_SCI_EN=0; CSME_SMI_EN=1;	CSME_SCI_STS CSME_SMI_STS
GPIO Lockdown Enable bit changes from '1' to '0'	No	Yes	GPIO_UNLOCK_SMI_EN=1	GPIO_UNLOCK_SMI_STS
USB 3.2 (xHCI) SMI Event	No	Yes	xHCI_SMI_EN=1	xHCI_SMI_STS
Wake Alarm Device Timer	Yes	Yes	WADT_EN	WADT_STS
ISH	Yes	No	ISH_EN	ISH_STS
RTC update-in-progress	No	Yes	Refer to Vol2	RTC_UIP_SMI_STS

Notes:
1. SCI_EN must be 1 to enable SCI, except for BIOS_RLS. SCI_EN must be 0 to enable SMI.
2. SCI can be routed to cause interrupt 9:11 or 20:23 (20:23 only available in APIC mode).
3. GBL_SMI_EN must be 1 to enable SMI.
4. EOS must be written to 1 to re-enable SMI for the next 1.
5. The processor must have SMI fully enabled when the processor is also enabled to trap cycles. If SMI is not enabled in conjunction with the trap enabling, then hardware behavior is undefined.
6. When a power button override first occurs, the system will transition immediately to S5. The SCI will only occur after the next wake to S0 if the residual status bit (PRBTNOR_STS) is not cleared prior to setting SCI_EN.
7. GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place.
8. Refer to General Purpose Input and Output on page 172 for specific GPIOs enabled for SCIs and/or SMIs
9. Secondary eSPI must assert SCI at least 100 us for the SCI event to be recognized.

PCI Express* SCI

PCI Express* ports and the processor have the ability to cause PME using messages. When a PME message is received, the processor will set the PCI_EXP_STS bit. If the PCI_EXP_EN bit is also set, the processor can cause an SCI using the GPE0_STS (replaced GPE1_STS) register.

PCI Express* Hot-Plug

PCI Express* has a hot-plug mechanism and is capable of generating a SCI using the GPE0 (replaced GPE1) register. It is also capable of generating an SMI. However, it is not capable of generating a wake event.

10.2.4 Sleep States

Sleep State Overview

The processor supports different sleep states S4/S5, which are entered by methods such as setting the SLP_EN bit or due to a Power Button press. The entry to the Sleep states is based on several assumptions:

- The G3 state cannot be entered using any software mechanism. The G3 state indicates a complete loss of power.

Initiating Sleep State

Sleep states (S4/S5) are initiated by:

- Masking interrupts, turning off all bus controller enable bits, setting the desired type in the SLP_TYP field, and then setting the SLP_EN bit. The hardware then attempts to gracefully put the system into the corresponding Sleep state.
- Pressing the PWRBTN# Signal for more than 4 seconds to cause a Power Button Override event. In this case the transition to the S5 state is less graceful, since there are no dependencies from the processor or on clocks other than the RTC clock.
- Assertion of the THERMTRIP# signal will cause a transition to the S5 state. This can occur when system is in the S0 state.
- Shutdown by integrated manageability functions (ASF/Intel® CSME).
- Internal watchdog timer timeout events.

Table 25. Sleep Types

Sleep Type	Comment
S4	The processor asserts SLP_S4#. The motherboard uses the SLP_S4# signal to shut off the power to the memory subsystem and any other unneeded subsystem. Only devices needed to wake from this state should be powered.
S5	The processor asserts SLP_S4# and SLP_S5#.

Exiting Sleep States

Sleep states (S4/S5) are exited based on wake events. The wake events forces the system to a full on state (S0), although some non-critical subsystems might still be shut off and have to be brought back manually. For example, the storage subsystem may be shut off during a sleep state and have to be enabled using a GPIO pin before it can be used.

Upon exit from the processor-controlled Sleep states, the WAK_STS bit is set. The possible causes of wake events (and their restrictions) are shown in the table below.

NOTE

If the BATLOW# signal is asserted, the processor does not attempt to wake from an S4/S5 state, even if the power button is pressed. This prevents the system from waking when the battery power is insufficient to wake the system. Wake events that occur while BATLOW# is asserted are latched by the processor, and the system wakes after BATLOW# is de-asserted.

Table 26. Causes of Wake Events

Cause	How Enabled	Wake from Sx	Wake from Sx After Power Loss[2]	Wake from "Reset" Types[3]
RTC Alarm	Set RTC_EN bit in PM1_EN_STS register.	Yes	Yes	No
Power Button	Always enabled as Wake event.	Yes	Yes	Yes

continued...

Cause	How Enabled	Wake from Sx	Wake from Sx After Power Loss[2]	Wake from "Reset" Types[3]
Any GPIOs except DSW GPIOs can be enabled for wake	Refer to Note 5	Yes	No	No
LAN_WAKE#	Enabled natively (unless pin is configured to be in GPIO mode)	Yes	Yes	Yes
Intel® High Definition Audio	Event sets PME_B0_STS bit; PM_B0_EN must be enabled. Cannot wake from S5 state if it was entered due to power failure or power button override.	Yes	Yes	No
Primary PME#	PME_B0_EN bit in GPE0_EN[127:96] register.	Yes	Yes	No
Secondary PME#	Set PME_EN bit in GPE0_EN[127:96] register.	Yes	Yes	No
PCI Express* WAKE# pin	PCIEXP_WAKE_DIS bit.	Yes	Yes	No
SMBALERT#	Refer to Note 4	Yes	Yes	Yes
Intel® CSME Non-Maskable Wake	Always enabled as a wake event.	Yes	Yes	Yes
Integrated WoL Enable Override	WoL Enable Override bit (in Configuration Space).	Yes	Yes	Yes
Wake Alarm Device	WADT_EN in GPE0_EN[127:96]	Yes	No	No

Notes:
1. If BATLOW# signal is low, processor will not attempt to wake from S4/S5 , even if a valid wake event occurs. This prevents the system from waking when battery power is insufficient to wake the system. However, once BATLOW# de-asserts, the system will boot.
2. This column represents what the processor would honor as wake events but there may be enabling dependencies on the device side which are not enabled after a power loss.
3. Reset Types include: Power Button override, Intel® CSME-initiated power button override, Intel® CSME-initiated host partition reset with power down, Intel® CSME Watchdog Timer, SMBus unconditional power down, processor thermal trip, processor catastrophic temperature event.
4. SMBALERT# signal is multiplexed with a GPIO pin that defaults to GPIO mode. Hence, SMBALERT# related wakes are possible only when this GPIO is configured in native mode, which means that BIOS must program this GPIO to operate in native mode before this wake is possible. Because GPIO configuration is in the resume well, wakes remain possible until one of the following occurs: BIOS changes the pin to GPIO mode, a G3 occurs .
5. There are only 72 bits in the GPE registers to be assigned to GPIOs, though any of the GPIOs can trigger a wake, only those status of GPIO mapped to 1-tier scheme are directly accessible through the GPE status registers. For those GPIO mapped under 2-tier scheme, their status would be reflected under single controller status, "GPIO_TIER2_SCI_STS" or GPE0_STS and further comparison needed to know which 2-tier GPI(s) has triggered the GPIO Tier 2 SCI.

PCI Express* WAKE# Signal and PME Event Message

PCI Express* ports can wake the platform from S4, S5 using the WAKE# pin. WAKE# is treated as a wake event, but does not cause any bits to go active in the GPE_STS register.

NOTE

PCI Express* WAKE# pin is an Output in S0ix states hence this pin cannot be used to wake up the system during S0ix states.

PCI Express* ports and the processor have the ability to cause PME using messages. These are logically OR'd to set the single PCI_EXP_STS bit. When a PME message is received, the processor will set the PCI_EXP_STS bit. If the PCI_EXP_EN bit is also set, the processor can cause an SCI via GPE0_STS register.

Sx-G3-Sx, Handling Power Failures

Depending on when the power failure occurs and how the system is designed, different transitions could occur due to a power failure.

The AFTERG3_EN bit provides the ability to program whether or not the system should boot once power returns after a power loss event. If the policy is to not boot, the system remains in an S5 state (unless previously in S4). There are only three possible events that will wake the system after a power failure.

Although PME_EN is in the RTC well, this signal cannot wake the system after a power loss. PME_EN is cleared by RTCRST#, and PME_STS is cleared by RSMRST#.

Table 27. Transitions Due to Power Failure

State at Power Failure	AFTERG3_EN Bit	Transition when Power Returns and BATLOW# is inactive
S0	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0

10.2.5 Event Input Signals and Their Usage

The processor has various input signals that trigger specific events. This section describes those signals and how they should be used.

PWRBTN# (Power Button)

The PWRBTN# signal operates as a "Fixed Power Button" as described in the *Advanced Configuration and Power Interface Specification.* PWRBTN# signal has a 16 ms de-bounce on the input. This logic ensures that presses for a duration less than 16ms are ignored. The minimum assertion duration that is guaranteed to be detected is 18ms. The state transition descriptions are included in the below table.

After any PWRBTN# assertion (falling edge), the 16 ms de-bounce applies before the state transition starts if PB_DB_MODE='0'. If PB_DB_MODE='1', the state transition starts right after any PWRBTN# assertion (before passing through the debounce logic) and subsequent falling PWRBTN# edges are ignored until after 16 ms.

During the time that any SLP_* signal is stretched for an enabled minimum assertion width, the host wake-up is held off. As a result, it is possible that the user will press and continue to hold the Power Button waiting for the system to wake. Unfortunately, a 4 second press of the Power Button is defined as an unconditional power down, resulting in the opposite behavior that the user was intending. Therefore, the Power Button Override Timer will be extended to 9-10 seconds while the SLP_* stretching timers are in progress. Once the stretching timers have expired, the Power Button will awake the system. If the user continues to press Power Button for the remainder of the 9-10 seconds it will result in the override condition to S5. Extension of the Power Button Override timer is only enforced following graceful sleep entry and during host partition resets with power cycle or power down. The timer is not extended immediately following power restoration after a global reset and G3.

The processor also supports modifying the length of time the Power Button must remain asserted before the unconditional power down occurs (4-14 seconds). The length of the Power Button override duration has no impact on the "extension" of the

power button override timer while SLP_* stretching is in progress. The extended power button override period while stretching is in progress remains 9-10 seconds in all cases.

Table 28. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI or SCI generated (depending on SCI_EN, PWRBTN_EN and GLB_SMI_EN)	Software typically initiates a Sleep state *Note:* Processing of transitions starts within 100 us of the PWRBTN# input pin to processor going low.[1]
S5	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup *Note:* Could be impacted by SLP_* min assertion. The minimum time the PWRBTN# pin should be asserted is 150 us. The processor will start processing this change once the minimum time requirement is satisfied.[1]
G3	PWRBTN# pressed	None	No effect since no power Not latched nor detected *Notes:* 1. During G3 exit, PWRBTN# pin must be kept de-asserted for a minimum time of 500 us after the RSMRST# has de-asserted.[2] 2. Beyond this point, the minimum time the PWRBTN# pin has to be asserted to be registered by processor as a valid wake event is 150 us.[1]
S0 – S4	PWRBTN# held low for at least 4 3 consecutive seconds	Unconditional transition to S5 state.	No dependence on processor or any other subsystem *Note:* Due to internal processor latency, it could take up to an additional ~1.3s after PWRBTN# has been held low for 4s before the system would begin transitioning to S5.
Notes: 1. If PM_CFG.PB_DB_MODE='0', the debounce logic adds 16 ms to the start/minimum time for processing of power button assertions. 2. This minimum time is independent of the PM_CFG.PB_DB_MODE value. 3. The amount of time PWRBTN# must be asserted is configurable via PM_CFG2.PBOP. 4 seconds is the default.			

Power Button Override Function

If PWRBTN# is observed active for at least four consecutive seconds (always sampled after the output from debounce logic), the processor should unconditionally transition to the G2/S5 state, regardless of present state (S0 – S4), even if the PLT_PWROK is not active. In this case, the transition to the G2/S5 state does not depend on any particular response from the processor, nor any similar dependency from any other subsystem.

The minimum period is configurable by BIOS and defaults to the legacy value of 4 seconds.

The PWRBTN# status is readable to check if the button is currently being pressed or has been released. If PM_CFG.PB_DB_MODE='0', the status is taken after the de-bounce. If PM_CFG.PB_DB_MODE='1', the status is taken before the de-bounce. In either case, the status is readable using the PWRBTN_LVL bit.

NOTE

The 4-second PWRBTN# assertion should only be used if a system lock-up has occurred.

Sleep Button

The *Advanced Configuration and Power Interface Specification* defines an optional Sleep button. It differs from the power button in that it only is a request to go from S0 to S4 (not S5). Also, in an S5 state, the Power Button can wake the system, but the Sleep Button cannot.

Although the processor does not include a specific signal designated as a Sleep Button, one of the GPIO signals can be used to create a "Control Method" Sleep Button. Refer to *Advanced Configuration and Power Interface Specification* for implementation details.

SYS_RESET# Signal

When the SYS_RESET# pin is detected as active (on signal's falling edge if de-bounce logic is disabled, or after 16 ms if 16 ms debounce logic is enabled), the processor attempts to perform a "graceful" reset by entering a host partition reset entry sequence.

Once the reset is asserted, it remains asserted for 5 to 6 ms regardless of whether the SYS_RESET# input remains asserted or not. It cannot occur again until SYS_RESET# has been detected inactive after the de-bounce logic, and the system is back to a full S0 state with PLTRST# inactive.

NOTES

1. The normal behavior for a SYS_RESET# assertion is host partition reset without power cycle. However, if bit 3 of the CF9h I/O register is set to '1' then SYS_RESET# will result in a full power-cycle reset.
2. It is not recommended to use the PLT_PWROK pin for a reset button as it triggers a global power cycle reset.
3. SYS_RESET# is in the primary power well but it only affects the system when PLT_PWROK is high.

THERMTRIP# Signal

If THERMTRIP# goes active, the processor is indicating an overheat condition, and the processor immediately transitions to an S5 state, driving SLP_S4#, SLP_S5# low, and setting the GEN_PMCON_2.PTS bit. The transition will generally look like a power button override.

When a THERMTRIP# event occurs, the processor will power down immediately without following the normal S0 -> S5 path. The processor will immediately drive SLP_S4#, and SLP_S5# low within 1 us after sampling THERMTRIP# active.

The reason the above is important is as follow: if the processor is running extremely hot and is heating up, it is possible (although very unlikely) that components around it, such as the processor, are no longer executing cycles properly. Therefore, if THERMTRIP# goes active, and the processor is relying on various handshakes to perform the power down, the handshakes may not be working, and the system will not power down. Hence the need for processor to power down immediately without following the normal S0 -> S5 path.

The processor provides filtering for short low glitches on the THERMTRIP# signal in order to prevent erroneous system shutdowns from noise. Glitches shorter than 25 nsec are ignored.

NOTE

A thermal trip event will clear the PWRBTN_STS bit.

Sx_Exit_Holdoff#

When S4/S5 is entered and SLP_A# is asserted, Sx_Exit_Holdoff# can be asserted by a platform component to delay resume to S0. SLP_A# de-assertion is an indication of the intent to resume to S0, but this will be delayed so long as Sx_Exit_Holdoff# is asserted. Sx_Exit_Holdoff is ignored outside of an S4/S5 entry sequence with SLP_A# asserted. With the de-assertion of RSMRST# (from G3->S0), this pin is a GPIO input and must be programmed by BIOS to operate as Sx_Exit_Holdoff. When SLP_A# is asserted (or it is de-asserted but Sx_Exit_Holdoff# is asserted), the processor will not access SPI Flash. How a platform uses this signal is platform specific.

Requirements to support Sx_Exit_Holdoff#

If the processor is in G3 or in the process of exiting G3 (RSMRST# is asserted), the EC must not allow RSMRST# to de-assert until the EC completed its flash accesses.

After the processor has booted up to S0 at least once since the last G3 exit, the EC can begin monitoring SLP_A# and using the SX_EXIT_HOLDOFF# pin to stop the processor from accessing flash. When SLP_A# asserts, if the EC intends to access flash, it will assert SX_EXIT_HOLDOFF#. To cover the case where the processor is going through a global reset, and not a graceful Sx+CMoff/Sx+CM3PG entry, the EC must monitor the SPI flash CS0# pin for 5 ms after SLP_A# assertion before making the determination that it is safe to access flash.

- If no flash activity is seen within this 5 ms window, the EC can begin accessing flash. Once its flash accesses are complete, the EC de-asserts (drives to ‘1’) SX_EXIT_HOLDOFF# to allow the processor to access flash.
- If flash activity is seen within this 5 ms window, the processor has gone through a global reset. And so the EC must wait until the processor reaches S0 again before re-attempting the holdoff flow.

NOTE

When eSPI is enabled, SX_EXIT_HOLDOFF# functionality is not available, and assertion of the signal will not impact Sx exit flows.

10.2.6 System Power Supplies, Planes, and Signals

Power Plane Control

The SLP_S4# or SLP_S5# output signal can be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done using the power supply, or by external FETs on the motherboard.

SLP_S5# output signal can be used to cut power to the system core supply.

SLP_A# output signal can be used to cut power to the Intel® Converged Security and Management Engine and SPI flash on a platform that supports the M3 state (for example, certain power policies in Intel® AMT).

SLP_LAN# output signal can be used to cut power to the external Intel® GbE PHY device.

SLP_S4# and SLP_S5# Sequencing

The system memory suspend voltage regulator is controlled by the Glue logic. The SLP_S4# signal should be used to remove power to system memory rather than the SLP_S5# signal. The SLP_S4# logic in the processor provides a mechanism to fully cycle the power to the DRAM and/or detect if the power is not cycled for a minimum time.

NOTE

To use the minimum DRAM power-down feature that is enabled by the SLP_S4# Assertion Stretch Enable bit (D31:F0:A4h Bit 3), the DRAM power must be controlled by the SLP_S4# signal.

PLT_PWROK Signal

When asserted, PLT_PWROK is an indication to the processor that its core well power rails are powered and stable. PLT_PWROK can be driven asynchronously. When PLT_PWROK is low, the processor asynchronously asserts PLTRST#. PLT_PWROK must not glitch, even if RSMRST# is low.

It is required that the power associated with PCIe* have been valid for 99 ms prior to PLT_PWROK assertion in order to comply with the 100 ms PCIe* 2.0 specification on PLTRST# de-assertion.

NOTE

SYS_RESET# is recommended for implementing the system reset button. This saves external logic that is needed if the PLT_PWROK input is used. Additionally, it allows for better handling of the SMBus and processor resets and avoids improperly reporting power failures.

BATLOW# (Battery Low)

The BATLOW# input can inhibit waking from S4, S5 if there is not sufficient power. It also causes an SMI if the system is already in an S0 state.

SLP_LAN# Pin Behavior

The processor controls the voltage rails into the external LAN PHY using the SLP_LAN# pin.

- The LAN PHY is always powered when the Host and Intel® CSME systems are running.
  - SLP_LAN#='1' whenever SLP_A#='1'.
- If the LAN PHY is required by Intel® CSME in Sx/M-Off, Intel® CSME must configure SLP_LAN#='1' irrespective of the power source and the destination power state. Intel® CSME must be powered at least once after G3 to configure this.
- If the LAN PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN.
- If the LAN PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN.
- If the LAN PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS.

NOTE

Intel® CSME configuration of SLP_LAN# in Sx/M-Off is dependent on Intel® CSME power policy configuration.

PRIMPWRDNACK Steady State Pin Behavior

Below table summarizes PRIMPWRDNACK pin behavior.

Table 29. PRIMPWRDNACK//GPP_A02 Pin Behavior

Pin	GPP_A02 Input/ Output (Determine by GP_IO_SEL bit)	Pin Value in S0	Pin Value in Sx/M-Off	Pin Value in Sx/M3
PRIMPWRDNACK	Native	0	Depends on Intel® CSME power package and power source (Note 1)	0
GPP_A02	IN	High-Z	High-Z	High-Z
	OUT	Depends on GPP_A02 output data value	Depends on GPP_A02 output data value	Depends on GPP_A02 output data value

Table 30. PRIMPWRDNACK During Reset

Reset Type (Note)	SPDA Value
Power-cycle Reset	0
Global Reset	0
Straight to S5	Processor initially drive '0' and then drive per Intel® CSME power policy configuration.
Note: Refer to Table 31 on page 89	

RTCRST# and SRTCRST#

RTCRST# is used to reset processor registers in the RTC Well to their default value. If a jumper is used on this pin, it should only be pulled low when system is in the G3 state and then replaced to the default jumper position. Upon booting, BIOS should recognize that RTCRST# was asserted and clear internal processor registers accordingly. It is imperative that this signal not be pulled low in the S0 to S5 states.

SRTCRST# is used to reset portions of the Intel® Converged Security and Management Engine and should not be connected to a jumper or button on the platform. The only time this signal gets asserted (driven low in combination with RTCRST#) should be when the coin cell battery is removed or not installed and the platform is in the G3 state. Pulling this signal low independently (without RTCRST# also being driven low) may cause the platform to enter an indeterminate state. Similar to RTCRST#, it is imperative that SRTCRST# not be pulled low in the S0 to S5 states.

10.2.7 Reset Behavior

When a reset is triggered, the processor completes any outstanding memory cycles and puts memory into a safe state before the platform is reset. When the processor is ready it asserts PLTRST#.

The processor does not require an acknowledge message from the processor to trigger PLTRST#. A global reset will occur after four seconds if an acknowledge from the processor is not received. When the processor causes a reset by asserting PLTRST#, its output signals will go to their reset states.

A reset in which the host platform is reset and PLTRST# is asserted is called a Host Reset or Host Partition Reset. Depending on the trigger a host reset may also result in power cycling, refer to the below table for details. If a host reset is triggered and the processor times out a Global Reset with power-cycle will occur.

A reset in which the host and Intel® CSME partitions of the platform are reset is called a Global Reset. During a Global Reset, all processor functionality is reset except RTC Power Well backed information and Suspend well status, configuration, and functional logic for controlling and reporting the reset. Intel® CSME and Host power back up after the power-cycle period.

Straight to S5 is another reset type where all power wells that are controlled by the SLP_S4#, and SLP_A# pins, as well as SLP_S5# and SLP_LAN# (if pins are not configured as GPIOs), are turned off. All processor functionality is reset except RTC Power Well backed information and Suspend well status, configuration, and functional logic for controlling and reporting the reset. The host stays there until a valid wake event occurs.

The following table shows the various reset triggers.

Table 31. Causes of Host and Global Resets

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
Write of 0Eh to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=0b	No	Yes	No[4]	
Write of 06h to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=0b	Yes	No	No[4]	

continued...

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
Write of 06h or 0Eh to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=1b	No	No	Yes	
SYS_RESET# Asserted and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No[4]	
SYS_RESET# Asserted and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No[4]	
SMBus Secondary Message received for Reset with Power-Cycle	No	Yes	No[4]	
SMBus Secondary Message received for Reset without Power-Cycle	Yes	No	No[4]	
SMBus Secondary Message received for unconditional Power Down	No	No	No	Yes
TCO Watchdog Timer reaches zero two times	Yes	No	No[4]	
Power Failure: PLT_PWROK signal goes inactive in S0	No	No	Yes	
SYS_PWROK Failure: SYS_PWROK signal goes inactive in S0	No	No	Yes	
Processor Thermal Trip (THERMTRIP#) causes transition to S5 and reset asserts	No	No	No	Yes
Processor internal thermal sensors signals a catastrophic temperature condition	No	No	No	Yes
Power Button 4 second override causes transition to S5 and reset asserts	No	No	No	Yes
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 1	No	No	Yes	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No[4]	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No[4]	
Intel® Converged Security and Management Engine Triggered Host Reset without Power-Cycle	Yes	No	No[4]	
Intel® Converged Security and Management Engine Triggered Host Reset with Power-Cycle	No	Yes	No[4]	
Intel® Converged Security and Management Engine Triggered Power Button Override	No	No	No	Yes
Intel® Converged Security and Management Engine Watchdog Timer Timeout	No	No	No[7]	Yes

continued...

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
Intel® Converged Security and Management Engine Triggered Global Reset	No	No	Yes	
Intel® Converged Security and Management Engine Triggered Host Reset with power down (host stays there)	No	Yes[5]	No[4]	
PLTRST# Entry Timeout (Note 6)	No	No	Yes	
PLT_PWROK Stuck Low	No	No	Yes	
Power Management Watchdog Timer	No	No	No[7]	Yes
Intel® Converged Security and Management Engine Hardware Uncorrectable Error	No	No	No[7]	Yes

Notes:
1. The processor drops this type of reset request if received while the system is in S4/S5.
2. Processor does not drop this type of reset request if received while system is in a software-entered S4/S5 state. However, the processor will perform the reset without executing the RESET_WARN protocol in these states.
3. The processor does not send warning message to processor, reset occurs without delay.
4. Trigger will result in Global Reset with Power-Cycle if the acknowledge message is not received by the processor.
5. The processor waits for enabled wake event to complete reset.
6. PLTRST# Entry Timeout is automatically initiated if the hardware detects that the PLTRST# sequence has not been completed within 4 seconds of being started.
7. Trigger will result in Global Reset with Power-Cycle if AGR_LS_EN=1 and Global Reset occurred while the current or destination state was S0.

10.3 Processor Graphics Power Management

10.3.1 Memory Power Savings Technologies

Intel® Rapid Memory Power Management (Intel® RMPM)

Intel® Rapid Memory Power Management (Intel® RMPM) conditionally places memory into self-refresh when the processor is in package C6 or deeper power state to allow the system to remain in the deeper power states longer for memory not reserved for graphics memory. Intel® RMPM functionality depends on graphics/display state (relevant only when processor graphics is being used), as well as memory traffic patterns generated by other connected I/O devices.

10.3.2 Display Power Savings Technologies

Intel® Seamless Display Refresh Rate Switching Technology (Intel® SDRRS Technology) with eDP* Port

Intel® DRRS provides a mechanism where the monitor is placed in a slower refresh rate (the rate at which the display is updated). The system is smart enough to know that the user is not displaying either 3D or media like a movie where specific refresh rates are required. The technology is very useful in an environment such as a plane where the user is in battery mode doing E-mail, or other standard office applications. It is also useful where the user may be viewing web pages or social media sites while in battery mode.

Intel® Display Power Saving Technology (Intel® DPST) 8.0

The Intel® DPST technique achieves back-light power savings while maintaining a good visual experience. This is accomplished by adaptively enhancing the displayed image while decreasing the back-light brightness simultaneously. The goal of this technique is to provide equivalent end-user-perceived image quality at a decreased back-light power level.

1. The original (input) image produced by the operating system or application is analyzed by the Intel® DPST subsystem. An interrupt to Intel® DPST software is generated whenever a meaningful change in the image attributes is detected. (A meaningful change is when the Intel® DPST determines if the brightness of the displaying images and the image enhancement and back-light control needs to be altered.)
2. Intel® DPST subsystem applies an image-specific enhancement to increase image brightness.
3. A corresponding decrease to the back-light brightness is applied simultaneously to produce an image with similar user-perceived quality (such as brightness) as the original image.

Intel® OLED Power Saving Technology (Intel® OPST) 1.1

Intel® OPST solution uses same HW infrastructure as Intel® DPST. Frames are processed using frame change threshold based interrupt mechanism similar to Intel® DPST. Intel® OPST SW algorithm determines which pixels in the frame should be dimmed to save power keeping visual quality (such as contrast, color) impact to acceptable level. Since there is no backlight for OLED panels, the power savings come solely from pixel dimming.

Panel Self-Refresh 2 (PSR 2) and Panel Replay (PR)

Panel Self-Refresh and Panel Replay (PR) features allow the Processor Graphics core to enter low-power state when the frame buffer content is not changing constantly. These features are available on panels capable of supporting Panel Self-Refresh or Panel Replay. PSR 2 adds partial frame updates and requires an compliant panel. Panel Replay adds further power optimizations by allowing refresh rate changes while in PR Active state.

Low-Power Single Pipe (LPSP)

Low-power single pipe is a power conservation feature that helps save power by keeping the inactive pipes powered OFF. LPSP is achieved by keeping a pipe enabled during eDP* only with minimal display pipeline support.

Low-Power Dual Pipe (LPDP)

This feature is similar to LPSP and is applicable for designs with dual eDP* panels.

Intel® Low Refresh Rate (Intel® LRR)

Intel ® LRR is a combination of PSR2 and Dynamic Refresh Rate Switching.

LRR uses two mechanisms for switching the refresh rate which are as follows:

- Pixel clock switching (Seamless DRRS/ DMRRS - Intel Specific)
- VTOTAL Change (VRR/Adaptive Sync - VESA Standard)

LRR is classified into different versions based on the RR switching technique, Intel platform support/capabilities, and eDP* panel support/capabilities.

Intel® Smart 2D Display Technology (Intel® S2DDT)

Intel® S2DDT reduces display refresh memory traffic by reducing memory reads required for display refresh. Power consumption is reduced by less accesses to the IMC. Intel S2DDT is only enabled in single pipe mode.

Intel® S2DDT is most effective with:

- Display images well suited to compression, such as text windows, slide shows, and so on. Poor examples are 3D games.
- Static screens such as screens with significant portions of the background showing 2D applications, processor benchmarks, and so on, or conditions when the processor is idle. Poor examples are full-screen 3D games and benchmarks that flip the display image at or near display refresh rates.

10.3.3 Processor Graphics Core Power Savings Technologies

Intel® Graphics Dynamic Frequency

Intel® Turbo Boost Technology 2.0 is the ability of the processor P/LP E cores and graphics (Graphics Dynamic Frequency) cores to opportunistically increase frequency and/or voltage above the guaranteed processor and graphics frequency for the given part. Intel® Graphics Dynamic Frequency is a performance feature that makes use of unused package power and thermals to increase application performance. The increase in frequency is determined by how much power and thermal budget is available in the package, and the application demand for additional processor or graphics performance. The processor P/LP E core control is maintained by an embedded controller. The graphics driver dynamically adjusts between P-States to maintain optimal performance, power, and thermals. The graphics driver will always place the graphics engine in its lowest possible P-State. Intel® Graphics Dynamic Frequency requires BIOS support. Additional power and thermal budget should be available.

Intel® Graphics Render Standby Technology (Intel® GRST)

Intel® Graphics Render Standby Technology is a technique designed to optimize the average power of the graphics part. The Graphics Render engine will be put in a sleep state, or Render Standby (RS), during times of inactivity or basic video modes. While in Render Standby state, the graphics part will place the VR (Voltage Regulator) into a low voltage state. Hardware will save the render context to the allocated context buffer when entering RS state and restore the render context upon exiting RS state.

Intel® Capped Frames Per Second (Intel® CFPS)

Intel® Capped Frames Per Second is a feature developed to save power during High FPS Gaming workloads while also achieving a tear and stutter free visual experience.

This feature ensures that the frame rate of the game does not exceed the panel refresh rate by matching screen updates to the Vertical Sync. That results fewer wakeups of graphics core and saves power.

When enabled, this feature works on any display panel, AC or DC mode and on any gaming workload.

10.4 TCSS Power States

Table 32. TCSS Power State

TCSS Power State	Processor PM State	Device Attached	Description
TC0	S0	Yes	xHCI, USB4 controllers may be active. USB4 DMA / PCIe may be active.
TC7	S*i2.1	Yes	xHCI is in D3. USB4 controller is in D3 or D0 idle. USB4 PCIe is inactive.
TC10	S*i2.2	No	Deepest Power state xHCI / USB4 controller are in D3. USB4 DMA / USB4 PCIe are in D3. IOM is in low power state.

Notes: 1. **S*i2.1/S*i2.2**, for more information, refer Naming Convention in Power Saving Features on page 77.
2. IOM - TCSS Input Output Manager:
3. The IOM interacts with the processor to perform power management, boot, reset, connect and disconnect devices to TYPE-C sub-system
4. TCSS Devices (xHCI / USB4 Controllers) - Power States:
   - D0 - Device at Active state.
   - D3 - Device at lowest-powered state.

10.5 Power and Performance Technologies

10.5.1 Intel® Thread Director

Intel® Thread Director helps monitor and analyze performance data in real time to seamlessly place the right application thread on the right core and optimize performance per watt.

Built directly into the hardware, Intel® Thread Director uses machine learning to schedule tasks on the right core at the right time (as opposed to relying on static rules). This helps to ensure that Performance-cores and Efficient-cores work in concert, background tasks do not slow you down, and you can have more applications open simultaneously.

- Monitors the runtime instruction mix of each thread and the state of each core with nanosecond precision.
- Provides runtime feedback to the OS to make the optimal decision for any workload.
- Dynamically adapts its guidance according to the Assured Based Power (ABP) of the system, operating conditions, and power settings.

10.5.2 Intel® Smart Cache Technology

The Intel® Smart Cache Technology is a shared Last Level Cache (LLC).

All IA cores are grouped into two clusters: Low Power (LP) cluster and Performance cluster

- LP cluster consists of a single module of 4 LP E-cores with shared L2 cache and doesn't have LLC.
- Performance cluster consists of several P-cores depend on particular SKU. Performance cluster has LLC that is shared between all its cores (of any type). The maximal size of LLC is 3MB (12 ways, set associative) per P-core.

Note: In case of odd number of P-Cores, the LLC size is 3MB x (#P-Cores + 1)

The LLC is non-inclusive.

The LLC may also be referred to as a 3rd level cache.

10.5.3 P-core LP E-core Level 0, Level 1 and Level 2 Caches

The 1st level cache is not shared between physical cores and each physical core has a separate set of caches.

The P-Core 1st level cache hierarchy is divided into:

- A Data Cache (DL0, DL1)
- An Instruction Cache (IL1)

On the data side, it is built as two-level cache, with L0 of 48KB and L1 of 192KB, both of which are 12-way set-associative.

On the instruction side, there is a single L1 cache of 64KB, which is 16-way set associative.

The LP E-Core 1st level cache hierarchy is divided into:

- A Data Cache (DL1)
- An Instruction Cache (IL1)

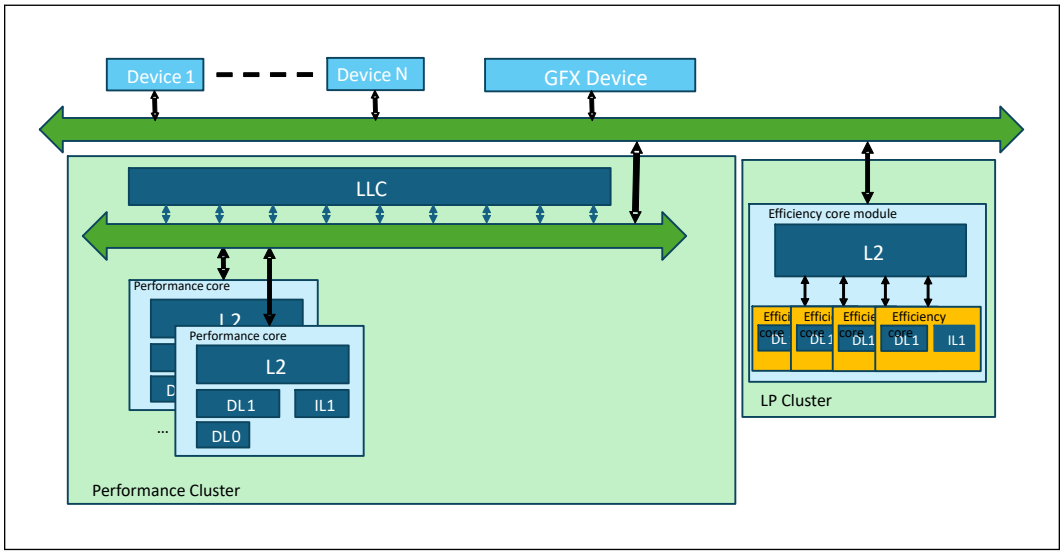
On the data side, it is built as one-level cache, with L1 of 32KB, 8-way set-associative.

On the instruction side, there is a single L1 cache of 64KB, which is 16-way set associative.

The 2nd level cache holds both data and instructions. It is also referred to as mid-level cache or MLC.

- The P-core 2nd level cache is not shared between physical cores and each physical core has a separate set of caches. Its size is 3MB and it is a 12-way associative non-inclusive cache.
- The LP E-core 2nd level cache is shared between physical cores across the Efficiency core module within the LP cluster. Its size is 4MB and it is a 16-way associative non-inclusive cache.

Figure 7. Intel® Core™ Processor (Series 3) Processor P-core and LP E-core Cache Hierarchy



NOTE

The above figure does not represent the exact number of cores.

Cache	P-core	LP E-core
L0 DL0	48KB 12-way set-associative per core	None
L1 DL1	192KB 12-way set-associative per core	32KB 8-way set-associative per core
L1 IL1	64KB 16-way set-associative per core	64KB 16-way set-associative per core
L2	3MB 12-way set-associative per core	4MB 16-way set-associative shared across Efficiency core module within LP Cluster (bundle of 4 LP E-cores)
L3	Maximum of 3 MB per physical core shared across Performance cluster[1]	None

Note: (1) In case of odd number of P-Cores, the LLC size is 3MB x (#P-Cores + 1)

10.5.4 Ring Interconnect

The Ring is a high speed, wide interconnect that links the processor P/LP E cores.

The Ring shares frequency and voltage with the Last Level Cache (LLC).

The Ring's frequency dynamically changes. Its frequency is relative to both processor cores and processor graphics frequencies.

10.5.5 Intel® Hybrid Technology

The processor contains two types of cores, denoted as Power and Efficient cores.

The Power (P) and Efficient (LP E) cores share the same instruction set and model specific registers (MSRs).

The available instruction sets, when hybrid computing is enabled, is limited compared to the instruction sets available to the P-core.

10.5.6 Intel® Turbo Boost Technology 2.0

The Intel® Turbo Boost Technology 2.0 allows the processor P/LP E core/processor graphics core to opportunistically and automatically run faster than the processor P/LP E core base frequency/processor graphics base frequency if it is operating below power, temperature, and current limits. The Intel® Turbo Boost Technology 2.0 feature is designed to increase the performance of both multi-threaded and single-threaded workloads.

Compared with previous generation products, Intel® Turbo Boost Technology 2.0 will increase the ratio of application power towards Processor Base Power and also allows to increase power above Processor Base Power as high as PL2 for short periods of time. Thus, thermal solutions and platform cooling that are designed to less than thermal design guidance might experience thermal and performance issues since more applications will tend to run at the maximum power limit for significant periods of time.

NOTE

Intel® Turbo Boost Technology 2.0 may not be available on all SKUs.

10.5.6.1 Intel® Turbo Boost Technology 2.0 Power Monitoring

When operating in turbo mode, the processor monitors its own power and adjusts the processor and graphics frequencies to maintain the average power within limits over a thermally significant time period. The processor estimates the package power for all components on the package. In the event that a workload causes the temperature to exceed program temperature limits, the processor will protect itself using the Adaptive Thermal Monitor.

10.5.6.2 Intel® Turbo Boost Technology 2.0 Power Control

Illustration of Intel® Turbo Boost Technology 2.0 power control is shown in the following sections and figures. Multiple controls operate simultaneously allowing customization for multiple systems thermal and power limitations. These controls allow for turbo optimizations within system constraints and are accessible using MSR, MMIO, and PECI mechanism.

10.5.6.3 Intel® Turbo Boost Technology 2.0 Frequency

To determine the highest performance frequency amongst active processor P/LP E cores, the processor takes the following into consideration:

- The number of processor P/LP E cores operating in the C0 state.
- The estimated processor P/LP E core current consumption and ICCMax settings.

- The estimated package prior and present power consumption and turbo power limits.
- The package temperature.

Any of these factors can affect the maximum frequency for a given workload. If the power, current, or thermal limit is reached, the processor will automatically reduce the frequency to stay within its Processor Base Power limit. Turbo processor frequencies are only active if the operating system is requesting the P0 state. For more information on P-states and C-states, refer to Power Management.

10.5.7 Intel® Adaptive Boost Technology

Intel® Adaptive Boost Technology (Intel® ABT) opportunistically increases the multicore turbo frequency while operating within IccMAX and temperature spec limitations.

Intel® ABT opportunistically delivers in-spec performance gains that are incremental to existing Turbo technologies. In systems equipped with performance spec power delivery, Intel® ABT allows additional multi-core turbo frequency while still operating within specified current and temperature limits.

10.5.8 Intel System Agent Enhanced SpeedStep ® Technology

Intel® System Agent Enhanced SpeedStep® Technology

Intel® System Agent consists of multiple IPs each providing dynamic voltage and frequency scaling capabilities. Intel SOCs scale voltage and frequency of the fabric and memory subsystem based on bandwidth demands and latency sensitivity of the workloads running on the SoCs.

Memory Geyserville (Memory GV): Memory subsystem provides four operating points for optimal memory subsystem power management. Memory GV targets memory controller and Memory PHY optimization by dynamically adjusting DDR data rates during light workload conditions when enabled. It also adjusts the memory subsystem operating points based on system power modes (such as best performance vs low battery modes).

Memory Training and Initialization: /MRC (Memory Reference Code) performs DDR training at maximum, mid, and minimum frequencies to establish optimal I/O and timing parameters for each operating point. To achieve optimal performance and memory power levels, the memory initialization and training process during first system boot, after CMOS clear, or following updates requires extended time compared to typical boot sequences. A black screen may be observed during this initialization and training process. Additional information on memory initialization processes is available in industry standard JEDEC Specifications at www.JEDEC.org.

Dynamic Frequency Scaling Operation:

Before changing DDR data rates, the processor places DDR memory into self-refresh mode and adjusts the necessary timing and voltage parameters to ensure stable operation at the new frequency, providing seamless transitions between memory performance states based on system workload demands.

10.5.9 Enhanced Intel SpeedStep® Technology

Enhanced Intel SpeedStep® Technology enables OS to control and select P-state. The following are the key features of Enhanced Intel SpeedStep® Technology:

- Multiple frequencies and voltage points for optimal performance and power efficiency. These operating points are known as P-states.
- Frequency selection is software controlled by writing to processor MSRs. The voltage is optimized based on the selected frequency and the number of active processors P/LP E cores.
  - Once the voltage is established, the PLL locks on to the target frequency.
  - All active processor P/LP E cores share the same frequency and voltage. In a multi-core processor, the highest frequency P-state requested among all active P/LP E cores is selected.
  - Software-requested transitions are accepted at any time. If a previous transition is in progress, the new transition is deferred until the previous transition is completed.
- The processor controls voltage ramp rates internally to ensure glitch-free transitions.

NOTE

Because there is low transition latency between P-states, a significant number of transitions per-second are possible.

10.5.10 Intel® Speed Shift Technology

Intel® Speed Shift Technology is an energy efficient method of frequency control by the hardware rather than relying on OS control. OS is aware of available hardware P-states and requests the desired P-state or it can let the hardware determine the P-state. The OS request is based on its workload requirements and awareness of processor capabilities. Processor decision is based on the different system constraints for example Workload demand, thermal limits while taking into consideration the minimum and maximum levels and activity window of performance requested by the Operating System.

For more details refer to:

- Intel® 64 Architectures Software Developer's Manual (SDM), Volume 3B.
- Appropriate BIOS Specification.
- Turbo Implementation Guide.

10.5.11 Intel® Advanced Vector Extensions 2 (Intel® AVX2)

Intel® Advanced Vector Extensions 2.0 (Intel® AVX2) is the latest expansion of the Intel instruction set. Intel® AVX2 extends the Intel® Advanced Vector Extensions (Intel® AVX) with 256-bit integer instructions, floating-point fused multiply-add (FMA) instructions, and gather operations. The 256-bit integer vectors benefit math, codec, image, and digital signal processing software. FMA improves performance in face detection, professional imaging, and high-performance computing. Gather operations increase vectorization opportunities for many applications. In addition to the vector

extensions, this generation of Intel processors adds bit manipulation instructions useful in compression, encryption, and general purpose software. For more information on Intel® AVX, refer to http://www.intel.com/software/avx

Intel® Advanced Vector Extensions (Intel® AVX) are designed to achieve higher throughput to certain integer and floating point operation. Due to varying processor power characteristics, utilizing AVX instructions may cause a) parts to operate below the base frequency b) some parts with Intel® Turbo Boost Technology 2.0 to not achieve any or maximum turbo frequencies. Performance varies depending on hardware, software and system configuration and you should consult your system manufacturer for more information.

Intel® Advanced Vector Extensions refers to Intel® AVX or Intel® AVX2.

For more information on Intel® AVX, refer to https://software.intel.com/en-us/isa-extensions/intel-avx.

NOTE

Intel® AVX and AVX2 Technologies may not be available on all SKUs.

10.5.11.1 AI Acceleration Extensions in Intel® AVX2

Vector Neural Network Instructions also known as VNNI or Intel® Deep Learning Boost are an extension that can help accelerate Deep learning workloads. The processor supports an AVX2 version of Vector Neural Network Instructions (AVX2 VNNI) which provides similar functionality as the AVX-512 VNNI instruction set but limited to AVX 2.Some platforms introduce support for all signed/unsigned combinations of operands.

For cases where the data input is in FP16 or BF16 data types, Some platforms add fast upconverts to FP32 so that the data can read from memory in FP16/BF16, computed in FP32 and down-converted back to FP16/BF16 for storage in memory.

10.5.12 Intel® 64 Architecture x2APIC

The x2APIC architecture extends the xAPIC architecture that provides key mechanisms for interrupt delivery. This extension is primarily intended to increase processor addressability.

Specifically, x2APIC:

- Retains all key elements of compatibility to the xAPIC architecture:
  - Delivery modes
  - Interrupt and processor priorities
  - Interrupt sources
  - Interrupt destination types
- Provides extensions to scale processor addressability for both the logical and physical destination modes
- Adds new features to enhance the performance of interrupt delivery
- Reduces the complexity of logical destination mode interrupt delivery on link based architectures

The key enhancements provided by the x2APIC architecture over xAPIC are the following:

- Support for two modes of operation to provide backward compatibility and extensibility for future platform innovations:
  - In xAPIC compatibility mode, APIC registers are accessed through memory mapped interface to a 4K-Byte page, identical to the xAPIC architecture.
  - In the x2APIC mode, APIC registers are accessed through the Model Specific Register (MSR) interfaces. In this mode, the x2APIC architecture provides significantly increased processor addressability and some enhancements on interrupt delivery.
- Increased range of processor addressability in x2APIC mode:
  - Physical xAPIC ID field increases from 8 bits to 32 bits, allowing for interrupt processor addressability up to 4G-1 processors in physical destination mode. A processor implementation of x2APIC architecture can support fewer than 32-bits in a software transparent fashion.
  - Logical xAPIC ID field increases from 8 bits to 32 bits. The 32-bit logical x2APIC ID is partitioned into two sub-fields – a 16-bit cluster ID and a 16-bit logical ID within the cluster. Consequently, ((2^20) - 16) processors can be addressed in logical destination mode. Processor implementations can support fewer than 16 bits in the cluster ID sub-field and logical ID sub-field in a software agnostic fashion.
- More efficient MSR interface to access APIC registers:
  - To enhance inter-processor and self-directed interrupt delivery as well as the ability to virtualize the local APIC, the APIC register set can be accessed only through MSR-based interfaces in x2APIC mode. The Memory Mapped IO (MMIO) interface used by xAPIC is not supported in x2APIC mode.
- The semantics for accessing APIC registers have been revised to simplify the programming of frequently-used APIC registers by system software. Specifically, the software semantics for using the Interrupt Command Register (ICR) and End Of Interrupt (EOI) registers have been modified to allow for more efficient delivery and dispatching of interrupts.
- The x2APIC extensions are made available to system software by enabling the local x2APIC unit in the "x2APIC" mode. To benefit from x2APIC capabilities, operating system support and a new BIOS are both needed, with special support for the x2APIC mode.
- The x2APIC architecture provides backward compatibility to the xAPIC architecture and forwards extensible for future Intel platform innovations.

NOTE

Intel® x2APIC Technology may not be available on all SKUs.

For more information, refer to Intel® *64 Architecture x2APIC Specification* at http://www.intel.com/products/processor/manuals/

10.5.13 Intel® Dynamic Tuning Technology (Intel® DTT)

Intel® Dynamic Tuning consists of a set of software drivers and applications that allow a system manufacturer to optimize system performance and usability by:

- Dynamically optimize turbo settings of IA processors, power and thermal states of the platform for optimal performance
- Dynamically adjust the processor's peak power based on the current power delivery capability for optimal system usability
- Dynamically mitigate radio frequency interference for better RF throughput.

10.5.14 Cache Line Write Back (CLWB)

Writes back to memory the cache line (if dirty) that contains the linear address specified with the memory operand from any level of the cache hierarchy in the cache coherence domain. The line may be retained in the cache hierarchy in the non-modified state. Retaining the line in the cache hierarchy is a performance optimization (treated as a hint by hardware) to reduce the possibility of a cache miss on a subsequent access. Hardware may choose to retain the line at any of the levels in the cache hierarchy, and in some cases, may invalidate the line from the cache hierarchy. The source operand is a byte memory location.

The CLWB instruction is documented in the Intel® Architecture Instruction Set Extensions Programming Reference (future architectures):

https://software.intel.com/sites/default/files/managed/b4/3a/319433-024.pdf

10.5.15 User Mode Wait Instructions

The *UMONITOR* and *UMWAIT* are user mode (Ring 3) instructions similar to the supervisor mode (Ring 0) *MONITOR*/*MWAIT* instructions without the C-state management capability.

TPAUSE is an enhanced *PAUSE* instruction.

The mnemonics for the three new instructions are:

- **UMONITOR**: operates just like MONITOR but allowed in all rings.
- **UMWAIT**: allowed in all rings, and no specification of target C-state.
- **TPAUSE**: similar to *PAUSE* but with a software-specified delay. Commonly used in spin loops.

10.6 Power and Internal Signals

10.6.1 Signal Description

Signal Name	Type	Description
GPP_V01/**ACPRESENT**	I	**ACPRESENT**: This input pin indicates when the platform is plugged into AC power or not. *Note:* An external pull-up resistor is required.
GPP_V00/**BATLOW#**	I	**Battery Low**: An input from the battery to indicate that there is insufficient power to boot the system. Assertion will prevent wake from S4/S5 states. This signal can also be enabled to cause an SMI# when asserted. *Note:* An external pull-up resistor is required.

continued...

Signal Name	Type	Description
GPP_V10/**LANPHYPC**	O	**LAN PHY Power Control**: LANPHYPC is used to indicate that power needs to be restored to the Platform LAN Connect Device.
PLT_PWROK	I	**PLT Power OK**: When asserted, it is an indication to the processor that all of its core power rails have been stable. The platform may drive asynchronously. For PLTRST# to be de-asserted PLT_PWROK must be asserted first - one of the conditions for PLTRST# de-assertion *Notes:* • Must not glitch, even if RSMRST# is low. • An external pull-down resistor is required. • Previously known as PCH_PWROK .
GPP_B13/**PLTRST#**	O	**Platform Reset**: The processor asserts PLTRST# to reset deviceson the platform. The processor asserts PLTRST# low in Sx states and when a cold, warm, or global reset occurs. The processor de-asserts PLTRST# upon exit from Sx states and the forementioned resets. There is no guaranteed minimum assertion time for PLTRST#.
GPP_V03/**PWRBTN#**	I	**Power Button**: The Power Button may cause an SMI# or SCI to indicate a system request to go to a sleep state. If the system is already in a sleep state, this signal will cause a wake event. If PWRBTN# is pressed for more than 4 seconds (default; timing is configurable), this will cause an unconditional transition (power button override) to the S5 state. Override will occur even if the system is in the S4 states. This signal has an internal Pull-up resistor and has an internal 16 ms de-bounce on the input.
RSMRST#	I	**Primary Well Reset**: This signal is used for resetting the primary power plane logic. This signal must be asserted for at least 10ms before de-asserting. *Note:* An external pull down resistor is required.
GPP_V06**/SLP_A#**	O	**SLP_A#**: Signal asserted when the Intel® CSME platform goes to M-Off or M3-PG. Depending on the platform, this pin may be used to control power to various devices that are part of the Intel® CSME sub-system in the platform. If you are not using SLP_A# for any functional purposes on your platform, or can tolerate lack of minimum assertion time, program the "SLP_A# minimum assertion width" value to the minimum. SLP_A# functionality can be utilized on the platform via either the physical pin or via the SLP_A# virtual wire over eSPI.
GPP_V11/**SLP_LAN#**	O	**LAN Sub-System Sleep Control**: When SLP_LAN# is de-asserted it indicates that the Platform LAN Connect Device must be powered. When SLP_LAN# is asserted, power can be shut off to the Platform LAN Connect Device. SLP_LAN# will always be de-asserted in S0 and anytime SLP_A# is de-asserted.
GPP_V04/**SLP_S3#**	O	**S3 Sleep Control:**SLP_S3# is for power plane control. This signal shuts off power to all non-critical systems when in the S4 or S5 state.
GPP_V05/**SLP_S4#**	O	**S4 Sleep Control**: SLP_S4# is for power plane control. This signal shuts power to all non-critical systems when in the S4 or S5 state. *Note:* This pin must be used to control the DRAM power in order to use the processor DRAM power-cycling feature.
GPP_V09/**SLP_S5#**	O	**S5 Sleep Control**: SLP_S5# is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 state.
GPP_V07/**SUSCLK**	O	**Suspend Clock**: This clock is a digitally buffered version of the RTC clock.
GPP_A02/ESPI_IO2/ **PRIMPWRDNACK**	O	**PRIMPWRDNACK**: Active high. Asserted by the processor on behalf of the Intel® CSME when it does not require the processor Primary well to be powered.
GPP_F09/ **SX_EXIT_HOLDOFF#**/ ISH_GP11	I	**Sx Exit Holdoff Delay**: Delay exit from Sx state after SLP_A# is de-asserted. *Note:* When eSPI is enabled, the flash sharing functionality using SX_EXIT_HOLDOFF# is not supported, but the pin still functions to hold off Sx exit after SLP_A# de-assertion.
SYS_RESET#	I	**System Reset**: This pin forces an internal reset after being de-bounced. *Note:* An external pull-up resistor is required.

continued...

Signal Name	Type	Description
GPP_E02/PROC_GP3/ **VRALERT#**/ISH_GP10	I	**VR Alert**: ICC Max throttling indicator from the processor voltage regulators. VRALERT# pin allows the VR to force processor throttling to prevent an over current shutdown. PMC based on the VRALERT# and messages from the processor. The messages from the processor allows the processor to constrain the processor to a particular power budget.
GPP_V12/**WAKE#**	I/OD	**PCI Express* Wake Event in Sx**: Input Pin in Sx. Sideband wake signal on PCI Express* asserted by components requesting wake up. *Notes:* • This is an output pin during S0ix states hence this pin cannot be used to wake up the system during S0ix states. • An external pull-up resistor is required.
EPD_ON_IN	I	Input signal to compute, this signal should be shorted with EPD_ON_OUT on the platform Level for test issue.
EPD_ON_OUT	O	Output signal from SOC tile, this signal should be shorted with EPD_ON_IN on the platform Level for test issue.
GPP_B12/**SLP_S0#**	O	**S0 Sleep Control**: When the processor is in C10 state, this pin will assert to indicate VR controller can go into a light load mode. This signal can also be connected to EC for other power management related optimizations.
SYS_PWROK	I	**System Power OK**: This generic power good input to the processor is driven and utilized in a platform-specific manner. While PLT_PWROK always indicates that the core wells of the processor are stable, SYS_PWROK is used to inform the processor that some other system component(s) power rails are stable, and the system is ready to start the exit from reset. *Note:* An external pull-down resistor is required.
VCCST_EN	O	Output signal from SOC tile to turn on the VCCST rail
PROC_C10_GATE#	O	When asserted, PROC_C10_GATE# is the indication to the system that the processor is entering C10.
GPP_B23/**TIME_SYNC1**/ ISH_GP6	I	**Time Synchronization**: Used for synchronization both input (latch time when pin asserted) and output (toggle pin when programmed time is hit).
VDD2PWRGOOD_IN	I	Power Good signal from processor. It needs to be connected to VDD2PWRGOOD_OUT on the platform.
VDD2PWRGOOD_OUT	O	Power Good signal from processor. It needs to be connected to VDD2PWRGOOD_IN on the platform.
PRIMACK#	I/O	Not in use, platform can leave unconnected.
GPP_E21/**PMCALERT#**	I	**PMC Alert Pin**: Supports USB-C* PD controller architecture. *Note:* An external pull-up resistor is required even if the signal is not used

10.6.2 Power Sequencing Signals

Table 33. Power Sequencing Signals

Signal Name	Description	Dir.	Buffer Type	Link Type
VIDSOUT	**VIDSOUT**, **VIDSCK, VIDALERT#:** These signals comprise a three-signal serial synchronous interface used to transfer power management information between the processor and the voltage regulator controllers.	I/O	I:GTL/ O:OD	SE
VIDSCK		O	OD	
VIDALERT#		I	CMOS	

10.6.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value
PWRBTN#	Pull-up	20 kohm +/- 30%
WAKE#	Pull-down	15 kohm - 40 kohm

10.6.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S4/S5
BATLOW#	Primary	Undriven	Undriven	Undriven
PROC_C10_GATE#	Primary	Driven High	Driven High	Driven High
LANPHYPC[10]	Primary	Undriven	Undriven	Undriven [7]
PLT_PWROK	RTC	Undriven	Undriven	Undriven
PLTRST#	Primary	Driven Low	Driven High	Driven Low
PWRBTN#	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up
RSMRST#	RTC	Undriven	Undriven	Undriven
SLP_A# [5]	Primary	Driven Low	Driven High	Driven High/Driven Low[12]
SLP_LAN# [5]	Primary	Driven Low	Driven Low	Driven High/Driven Low[7]
SLP_S0# [1]	Primary	Driven High	Driven High	Driven High
SLP_S3# [5]	Primary	Driven Low	Driven High	Driven Low
SLP_S4# [5]	Primary	Driven Low	Driven High	Driven Low
SLP_S5# [5]	Primary	Driven Low	Driven High	Driven High/Driven Low[3]
SUSCLK [7,10]	Primary	Driven Low	Toggling	Toggling
PRIMPWRDNACK [7,10]	Primary	Driven Low	Driven Low	Driven Low[4]
SX_EXIT_HOLDOFF# [9]	Primary	Undriven	Undriven	Undriven
SYS_PWROK	Primary	Undriven	Undriven	Undriven
SYS_RESET#	Primary	Undriven	Undriven	Undriven

continued...

Signal Name	Power Plane	During Reset	Immediately after Reset	S4/S5
VRALERT# [9]	Primary	Undriven	Undriven	Undriven
WAKE# [10]	Primary	Undriven	Undriven	Undriven

Notes:
1. Driven High during S0 and driven Low during S0i3 when all criteria for assertion are met.
2. SLP_S4# is driven low in S4/S5.
3. SLP_S5# is driven high in S4, driven low in S5.
4. .PRIMPWRDNACK is always '0' while in M0 or M3, but can be driven to '0' or '1' while in Moff state. PRIMPWRDNACK is the default mode of operation.
5. The pad should only be pulled low momentarily when the corresponding buffer power supply is not stable.
6. Based on wake event and Intel® CSME state.
7. Internal weak pull-down resistor is enabled during power sequencing.
8. Pin state is a function of whether the platform is configured to have Intel® CSME on or off in Sx.
9. Output High-Z, not glitch free.
10. Output High-Z

11.0 Power Delivery

11.1 Power and Ground Signals

This section describes the processor power rails.

Table 34. Power Rail Descriptions

Signal Name	Description
VSS	Ground
Processor Rails	
VCC_PCORE	Dynamic SVID power rail to support processor P-Cores.
VCC_LP_ECORE	Dynamic SVID power rail to support processor LP E-Cores.
VCCGT	Dynamic SVID power rail to support processor Graphics.
VCCSA	Dynamic SVID power rail to support processor NPU, Display Engine, Media, Memory Controller, IPU.
VCCPRIM_VNNAON	Fixed power rail to support digital blocks.
VCCPRIM_VNNAON_FLTR	Fixed VCCPRIM_VNNAON power rail with filter requirements.
VCCPRIM_UCIE_ANA	Dedicated fixed power rail to support UCIE analog.
VCCPRIM_IO	Fixed power rail to support analog blocks.
VCCPRIM_IO_FLTR	Fixed VCCPRIM_IO power rail with filter requirements.
VCCRTC	Fixed power rail from coin-cell battery to support RTC (Real Time Clock).
VCCST	Fixed power rail to support digital and analog blocks.
VCCPRIM_3P3	Fixed 3.3V for primary well.
VCCPRIM_1P8	Fixed 1.8V for primary well.
VCCPRIM_1P8_FLTR	Fixed 1.8V primary well with filter requirements.
Memory Rails	
VDDQ	Fixed power rail to support Processor and DRAM.
VDD2	Fixed power rail to support Processor and DRAM.

Table 35. Power Rail Sense Signals

Signal Name	Description
Sense Signals	
VCC_PCORE_SENSE	VCC_PCORE sense
VCC_LP_ECORE_SENSE	VCC_ECORE sense
VCCGT_SENSE	VCCGT sense
VCCSA_SENSE	VCCSA sense
VCCPRIM_IO_SENSE	VCCPRIM_IO sense
Ground Sense Signals	
VCC_PCORE_VSS_SENSE	VCC_PCORE ground sense
VCC_LP_ECORE_VSS_SENSE	VCC_ECORE ground sense
VCCGT_VSS_SENSE	VCCGT ground sense
VCCSA_VSS_SENSE	VCCSA ground sense

11.2 Digital Linear Voltage Regulator (DLVR)

Digital Linear Voltage Regulator (DLVR) is implemented on Processor internal power rails (VCC_CORE and VCCSA) for power saving, by gating power for Cores and digital IPs. DLVR mitigate EMI/RFI using Spread Spectrum Clock (SSC).

11.3 Current Excursion Protection (CEP)

This power management is a Processor integrated detector which senses when the Processor load current exceeds a preset threshold by monitoring for a Processor power domain voltage droop at the Processor power domain IMVPVR sense point. The Processor compares the IMVPVR output voltage with a preset threshold voltage (VTRIP) and when the IMVPVR output voltage is equal to or less than VTRIP , the Processor internally throttles itself to reduce the Processor load current and the power.

IMVP9.3 VRs enhance the CEP detector by adding a cycle by cycle current limiting feature where the IMVPVR quickly enters cycle by cycle current limit (becomes a current source) with the VR output current limited to a preset value (ITRIP) as set in the ICC_limit register.

11.4 Fast V-Mode (FVM)

This power management feature protects VR FETs and inductors from experiencing the full per-rail ICCMAX current, while also shielding upstream input power devices from the full power load.

IccMAX.APP represents the actual maximum current expected during real workloads when FVM is enabled on a per-rail basis, which is lower than IccMAX (the current when FVM is disabled).

Fast V-mode enables platform power subsystems to be designed around IccMAX.APP rather than IccMAX on a per-VR basis, while delivering better performance compared to proactive ICCMAX reduction approaches.

FVM is enabled in SVID power rails (per SKU), refer to Electrical Specification for exact configuration.

11.5 Vsys_crit based Reactive PL4 with PL4 Boost

2S (two cells in series) battery systems, while being efficient in power conversion, are at risk of "brownout" during peak power events, hence they tend to request lower PL4 levels. This PL4 level fluctuates depending on the remaining state of charge (RSOC) of the battery.

The system can implement the Reactive PL4 mechanism called "PL4 Boost" given the:

1. Effective capacitance on V_{SYS}
2. Power removal reaction speed due to a system rail undervoltage event.

The Processor uses PL4 Boost to calculate a higher performance frequency with a potentially higher P_{max} than the programmed PL4 value. Upon IMVP FORCEPR# assertion, the programmed PL4 level is respected. Oscillatory assertions are addressed when identified.

The PL4 Boost feature enables higher peak performance and/or responsiveness for 2S battery systems in low remaining state of charge (RSOC) conditions. Responsiveness gains are a result of the Processor using higher frequency states while having a reactive mechanism in place to quickly reduce loading.

Using 2S batteries allows for the most efficient power conversion and battery density per volume versus 3S batteries, however, in low RSOC conditions there is risk of brownouts due to system rail voltage droop when using high PL4 setting.

11.6 Thermally Equal Turbo-boost Algorithm (ThETA)

The Thermally Equal Turbo-boost Algorithm (ThETA) is a new feature that controls battery discharge current in DC mode.

Battery discharge current (known as Isys or Ibatt) refers to the current flowing from the battery to the system when operating in DC mode.

12.0 Thermal Management

Table 36. Definitions/Acronyms

Acronyms	Description
Max Operating Temperature	This is the maximum operating temperature allowed as reported by temperature sensors. Instantaneous temperature may exceed this value for short durations. *Note:* Maximum observable temperature is configurable by system vendor and can be design specific.

12.1 Processor Thermal Management

The thermal solution provides both component-level and system-level thermal management. To allow optimal operation and long-term reliability of Intel processor-based systems, the system/processor thermal solution should be designed so that the processor:

- Remains below the maximum operating temperature specification at the maximum Processor Base Power.
- Conforms to system constraints, such as system acoustics, system skin-temperatures, and exhaust-temperature requirements.

CAUTION

Thermal specifications given in this chapter are on the component and package level and apply specifically to the processor. Operating the processor outside the specified limits may result in permanent damage to the processor and potentially other components in the system.

12.1.1 Thermal Considerations

The Processor Base Power as is the maximum sustained power that should be used for the design of the processor thermal solution. Processor Base Power is a power dissipation and operating temperature condition limit, specified in this document, that is validated during manufacturing for the base configuration when executing a near worst case commercially available workload as specified by Intel for the SKU segment. Processor Base Power may be exceeded for short periods of time or if running a very high power workload.

The processor integrates multiple processing IA cores, graphics cores and for some SKUs a chipset on a single package. This may result in power distribution differences across the package and should be considered when designing the thermal solution.

Intel ®Turbo Boost Technology 2.0 allows processor IA cores to run faster than the base frequency. It is invoked opportunistically and automatically as long as the processor is conforming to its temperature, power delivery, and current control limits. When Intel® Turbo Boost Technology 2.0 is enabled:

- Applications are expected to run closer to Processor Base Power more often as the processor will attempt to maximize performance by taking advantage of estimated available energy budget in the processor package.
- The processor may exceed the Processor Base Power for short durations to utilize any available thermal capacitance within the thermal solution. The duration and time of such operation can be limited by platform runtime configurable registers within the processor.
- Graphics peak frequency operation is based on the assumption of only one of the graphics domains (GT/GTx) being active. This definition is similar to the IA core Turbo concept, where peak turbo frequency can be achieved when only one IA core is active. Depending on the workload being applied and the distribution across the graphics domains the user may not observe peak graphics frequency for a given workload or benchmark.
- Thermal solutions and platform cooling that is designed to less than thermal design guidance may experience thermal and performance issues.

NOTE

Intel® Turbo Boost Technology 2.0 availability may vary between the different SKUs.

12.1.1.1 Package Power Control

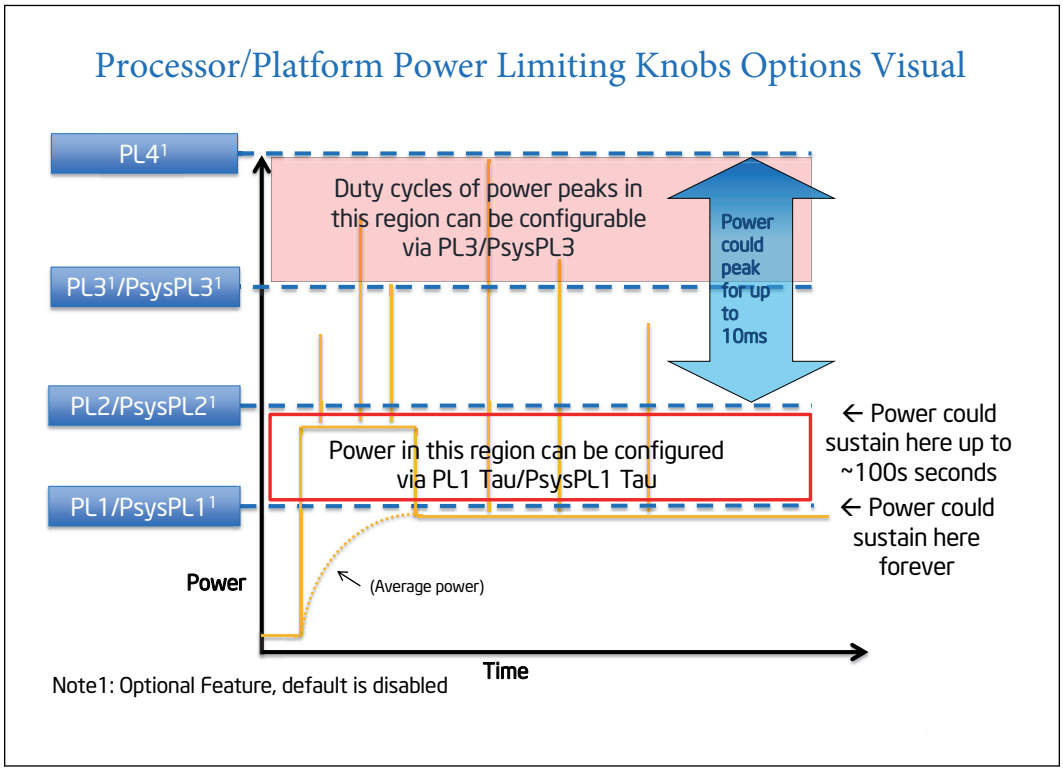
The package power control settings of PL1, PL2, PL3, PL4, and Tau allow the designer to configure Intel® Turbo Boost Technology 2.0 to match the platform power delivery and package thermal solution limitations.

- **Power Limit 1 (PL1)**: A threshold for average power that will not exceed - recommend to set to equal Processor Base Power. PL1 should not be set higher than thermal solution cooling limits.
- **Power Limit 2 (PL2)**: A threshold that if exceeded, the PL2 rapid power limiting algorithms will attempt to limit the spike above PL2.
- **Power Limit 3 (PL3)**: A threshold that if exceeded, the PL3 rapid power limiting algorithms will attempt to limit the duty cycle of spikes above PL3 by reactively limiting frequency. This is an optional setting
- **Power Limit 4 (PL4)**: A limit that will not be exceeded, the PL4 power limiting algorithms will preemptively limit frequency to prevent spikes above PL4.
- **Turbo Time Parameter (Tau)**: An averaging constant used for PL1 exponential weighted moving average (EWMA) power calculation.

NOTES

1. Implementation of Intel® Turbo Boost Technology 2.0 only requires configuring PL1, PL1, Tau and PL2.
2. PL3 and PL4 are disabled by default.
3. Intel® Dynamic Tuning Technology (DTT) is recommended for performance improvement in platforms. Dynamic Tuning is configured by system manufacturers dynamically optimizing the processor power based on the current platform thermal and power delivery conditions. Contact Intel Representatives for enabling details.

Figure 8. Package Power Control



12.1.1.2 Platform Power Control

The processor introduces Psys (Platform Power) to enhance processor power management. The Psys signal needs to be sourced from a compatible charger circuit and routed to the IMVP (voltage regulator). This signal will provide the total thermally relevant platform power consumption (processor and rest of platform) via SVID to the processor.

When the Psys signal is properly implemented, the system designer can utilize the package power control settings of PsysPL1/Tau, PsysPL2, and PsysPL3 for additional manageability to match the platform power delivery and platform thermal solution limitations for Intel® Turbo Boost Technology 2.0. The operation of the PsysPL1/tau, PsysPL2 and PsysPL3 are analogous to the processor power limits described in Package Power Control on page 111.

- **Platform Power Limit 1 (PsysPL1)**: A threshold for average platform power that will not be exceeded - recommend to set to equal platform thermal capability.
- **Platform Power Limit 2 (PsysPL2)**: A threshold that if exceeded, the PsysPL2 rapid power limiting algorithms will attempt to limit the spikes above PsysPL2.
- **Platform Power Limit 3 (PsysPL3)**: A threshold that if exceeded, the PsysPL3 rapid power limiting algorithms will attempt to limit the duty cycle of spikes above PsysPL3 by reactively limiting frequency.
- **PsysPL1 Tau**: An averaging constant used for PsysPL1 exponential weighted moving average (EWMA) power calculation.
- The Psys signal and associated power limits / Tau are optional for the system designer and disabled by default.
- The Psys data will not include power consumption for charging.
-
- The Intel Dynamic Tuning (DTT/DPTF) is recommended for performance improvement in mobile platforms. Dynamic Tuning is configured by system manufacturers dynamically optimizing the processor power based on the current platform thermal and power delivery conditions. Contact Intel Representatives for enabling details.

12.1.1.3 Turbo Time Parameter (Tau)

Turbo Time Parameter (Tau) is a mathematical parameter (units of seconds) that controls the Intel® Turbo Boost Technology 2.0 algorithm. During a maximum power turbo event, the processor could sustain PL2 for a duration longer than the Turbo Time Parameter. If the power value and/or Turbo Time Parameter is changed during runtime, it may take some time based on the new Turbo Time Parameter level for the algorithm to settle at the new control limits. The time varies depending on the magnitude of the change, power limits and other factors. There is an individual Turbo Time Parameter associated with Package Power Control and Platform Power Control.

12.1.2 Thermal Management Features

Occasionally the processor may operate in conditions that are near to its maximum operating temperature. This can be due to internal overheating or overheating within the platform. In order to protect the processor and the platform from thermal failure, several thermal management features exist to reduce package power consumption and thereby temperature in order to remain within normal operating limits. Furthermore, the processor supports several methods to reduce memory power.

12.1.2.1 Skin Temperature Control (STC)

Skin Temperature Control (STC) is a SoC Thermal Management feature that uses a Machine Learning algorithm to automatically manage system power and temperatures. STC tuning requires providing a target skin temperature and specifying an overshoot allowance.

The system continuously learns and adjusts throttling behaviors similar to DTT's Intelligent Thermal Management policy, operating independently or in coordination with system software through programmable PECI and MMIO configured via EC or BIOS.

STC provides three programmable independent temperature thresholds per domain (PECI and MMIO), configurable sensor settings, telemetry debug capabilities through PMT to read PECI data and status, and support for software-based sensors enabled through MMIO that require OS runtime updates. By default, STC utilizes PECI to receive updated sensors temperature. SW sensor overrides can be used, but the MMIO domain temperatures must be updated by SW.

As an always-available, hardware-based solution that works across different operating systems, STC operates during boot, OS installation, in UEFI shell or when thermal management drivers are absent, providing thermal backup for skin temperature management and enabling tighter interoperability in high concurrency scenarios.

12.1.2.2 Adaptive Thermal Monitor

The purpose of the Adaptive Thermal Monitor is to reduce processor IA core power consumption and temperature until it operates below its maximum operating temperature. Processor IA core power reduction is achieved by:

- Adjusting the operating frequency (using the processor IA core ratio multiplier) and voltage.
- Modulating (starting and stopping) the internal processor IA core clocks (duty cycle).

The Adaptive Thermal Monitor can be activated when the package temperature, monitored by any Digital Thermal Sensor (DTS), meets its maximum operating temperature. The maximum operating temperature implies Maximum Operating Temperature.

Reaching the maximum operating temperature activates the Thermal Control Circuit (TCC). When activated the TCC causes both the processor IA core and graphics core to reduce frequency and voltage adaptively. The Adaptive Thermal Monitor will remain active as long as the package temperature remains at its specified limit. Therefore, the Adaptive Thermal Monitor will continue to reduce the package frequency and voltage until the TCC is de-activated.

Maximum Operating Temperature is factory calibrated and is not user configurable. The default value is software visible in the TEMPERATURE_TARGET (1A2h) MSR, bits [23:16].

The Adaptive Thermal Monitor does not require any additional hardware, software drivers, or interrupt handling routines. It is not intended as a mechanism to maintain processor thermal control to PL1 = Processor Base Power. The system design should provide a thermal solution that can maintain normal operation when PL1 = Processor Base Power within the intended usage range.

Adaptive Thermal Monitor protection is always enabled.

TCC Activation Offset

TCC Activation Offset can be set as an offset from Maximum Operating Temperature to lower the onset of TCC and Adaptive Thermal Monitor.

Intel recommends maintaining the default TCC Activation Offset = 0 setting to ensure maximum system performance while preserving quality and reliability.

In addition, there is an optional time window (Tau) to manage processor performance at the TCC Activation offset value via an EWMA (Exponential Weighted Moving Average) of temperature.

TCC Activation Offset with Tau=0

An offset (degrees Celsius) can be written to the TEMPERATURE_TARGET (1A2h) MSR, bits [29:24], the offset value will be subtracted from the value found in bits [23:16]. When the time window (Tau) is set to zero, there will be no averaging, the offset, will be subtracted from the Maximum Operating Temperature value and used as a new maximum temperature set point for Adaptive Thermal Monitoring. This will have the same behavior as in prior products to have TCC activation and Adaptive Thermal Monitor to occur at this lower target silicon temperature.

If enabled, the offset should be set lower than any other passive protection such as ACPI _PSV trip points.

Frequency / Voltage Control

Upon Adaptive Thermal Monitor activation, the processor attempts to dynamically reduce processor temperature by lowering the frequency and voltage operating point. The operating points are automatically calculated by the processor IA core itself and do not require the BIOS to program them as with previous generations of Intel processors. The processor IA core will scale the operating points such that:

- The voltage will be optimized according to the temperature, the processor IA core bus ratio and the number of processor IA cores in deep C-states.
- The processor IA core power and temperature are reduced while minimizing performance degradation.

Once the temperature has dropped below the trigger temperature, the operating frequency and voltage will transition back to the normal system operating point.

Once a target frequency/bus ratio is resolved, the processor IA core will transition to the new target automatically.

- On an upward operating point transition, the voltage transition precedes the frequency transition.
- On a downward transition, the frequency transition precedes the voltage transition.
- The processor continues to execute instructions. However, the processor will halt instruction execution for frequency transitions.

If a processor load-based Enhanced Intel SpeedStep Technology/P-state transition (through MSR write) is initiated while the Adaptive Thermal Monitor is active, there are two possible outcomes:

- If the P-state target frequency is higher than the processor IA core optimized target frequency, the P-state transition will be deferred until the thermal event has been completed.
- If the P-state target frequency is lower than the processor IA core optimized target frequency, the processor will transition to the P-state operating point.

Clock Modulation

If the frequency/voltage changes are unable to end an Adaptive Thermal Monitor event, the Adaptive Thermal Monitor will utilize clock modulation. Clock modulation is done by alternately turning the clocks off and on at a duty cycle (ratio between clock "on" time and total time) specific to the processor. The duty cycle is factory configured to 25% on and 75% off and cannot be modified. The period of the duty cycle is configured to 32 microseconds when the Adaptive Thermal Monitor is active. Cycle times are independent of processor frequency. A small amount of hysteresis has been included to prevent excessive clock modulation when the processor temperature is near its Maximum Operating Temperature. Once the temperature has dropped below the Maximum Operating Temperature, and the hysteresis timer has expired, the Adaptive Thermal Monitor goes inactive and clock modulation ceases. Clock modulation is automatically engaged as part of the Adaptive Thermal Monitor activation when the frequency/voltage targets are at their minimum settings. Processor performance will be decreased when clock modulation is active. Snooping and interrupt processing are performed in the normal manner while the Adaptive Thermal Monitor is active.

Clock modulation will not be activated by the Package average temperature control mechanism.

Thermal Throttling

As the processor approaches Maximum Operating Temperature, a throttling mechanism will engage to protect the processor from over-heating and provide control thermal budgets.

Achieving this is done by reducing IA and other subsystem agent's voltages and frequencies in a gradual and coordinated manner that varies depending on the dynamics of the situation. IA frequencies and voltages will be directed down as low as LFM (Lowest Frequency Mode), each E-core module (4 E-cores) or each P-core can be thermally throttle independently. Further restricts are possible via Thermal Threshold point (TT1) under conditions where thermal budget cannot be re-gained fast enough with voltages and frequencies reduction alone. TT1 keeps the same processor voltage and clock frequencies the same yet skips clock edges to produce effectively slower clocking rates. This will effectively result in observed frequencies below LFM on the Windows PERF monitor.

12.1.2.3 Digital Thermal Sensor

Each processor has multiple on-tile Digital Thermal Sensor (DTS) that detects the instantaneous temperature of processor IA, GT and other areas of interest.

Temperature values from the DTS can be retrieved through:

- A software interface using processor Model Specific Register (MSR).
- A processor hardware interface.

When the temperature is retrieved by the processor MSR, it is the instantaneous temperature of the given DTS. When the temperature is retrieved using PECI, it is the average of the highest DTS temperature in the package over a 256 ms time window. Intel recommends using the PECI reported temperature for platform thermal control that benefits from averaging, such as fan speed control. The average DTS temperature may not be a good indicator of package Adaptive Thermal Monitor activation or rapid increases in temperature that triggers the Out of Specification status bit within the PACKAGE_THERM_STATUS (1B1h) MSR and IA32_THERM_STATUS (19Ch) MSR.

Code execution is halted in C1 or deeper C-states. Package temperature can still be monitored through PECI in lower C-states.

Unlike traditional thermal devices, the DTS outputs a temperature relative to the maximum supported operating temperature of the processor, regardless of TCC activation offset. It is the responsibility of software to convert the relative temperature to an absolute temperature. The absolute reference temperature is readable in the TEMPERATURE_TARGET (1A2h) MSR. The temperature returned by the DTS is an implied negative integer indicating the relative offset from Maximum Operating Temperature. The DTS does not report temperatures greater than Maximum Operating Temperature. The DTS-relative temperature readout directly impacts the Adaptive Thermal Monitor trigger point. When a package DTS indicates that it has reached the TCC activation (a reading of 0h, except when the TCC activation offset is changed), the TCC will activate and indicate an Adaptive Thermal Monitor event. A TCC activation will lower both processor IA core and graphics core frequency, voltage, or both. Changes to the temperature can be detected using two programmable thresholds located in the processor thermal MSRs. These thresholds have the capability of generating interrupts using the processor IA core's local APIC. Refer to the *Intel 64 Architectures Software Developer's Manual* for specific register and programming details.

Fan Speed Control with Digital Thermal Sensor

Digital Thermal Sensor based fan speed control (T_{FAN}) is a recommended feature to achieve optimal thermal performance. At the T_{FAN} temperature, Intel recommends full cooling capability before the DTS reading reaches Maximum Operating Temperature.

12.1.2.4 FORCEPR# Signal

The FORCEPR# is an input signal to the CPU. It is used to reduce processor's electrical load following power and thermal events. FORCEPR# should not be used as an output indication.

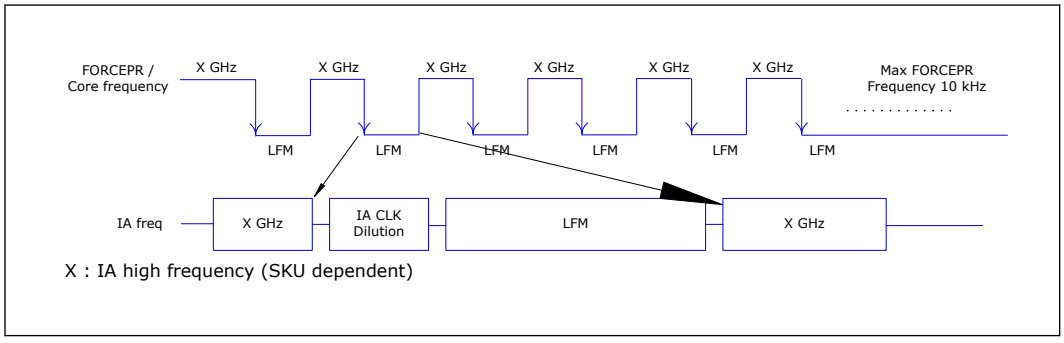
The FORCEPR# signal can be configured to the following mode:

- Input Only: FORCEPR# is driven by an external force.

Following FORCEPR# assertion, Fast FORCEPR# feature will instantly reduce frequency and CPU will limit the frequency to lowest frequency according to FORCEPR_RESPONSE configuration that will remain until FORCEPR# de-assertion.

FORCEPR# Demotion Algorithm will be activated in case of multiple FORCEPR# events.

Figure 9. FORCEPR# Demotion Description



12.1.2.5 FORCEPR Demotion

FORCEPR# demotion algorithm is designed to improve system performance following multiple Platform FORCEPR# consecutive assertions. When detecting several FORCEPR# consecutive assertions the processor will reduce the max frequency in order to reduce the FORCEPR# assertions events. The processor will keep reducing the frequency until no consecutive assertions detected. The processor will raise the frequency if no consecutive FORCEPR# assertion events will occur.

12.1.2.6 Voltage Regulator Protection using FORCEPR#

FORCEPR# may be used for thermal protection of voltage regulators (VR). System designers can create a circuit to monitor VR temperature and assert FORCEPR# to activate the TCC when the VR temperature limit is reached.

When FORCEPR# is configured as an input only signal, system assertion of FORCEPR# is recognized by the processor results in power reduction. Power reduction scales down to LFM for the duration of the platform FORCEPR# assertion and is supported by both the processor IA cores and graphics cores.

Systems should still provide proper cooling for the VR and rely on bi-directional FORCEPR# only as a backup protection in case of system cooling failure. Overall, the system thermal design should allow the power delivery circuitry to operate within its temperature specification even while the processor is operating at its maximum performance.

Adaptive Thermal Monitor protection is always enabled.

NOTE

During FORCEPR# demotion, the core frequency may be reduced below LFM for several uSec.

12.1.2.7 Thermal Solution Design and FORCEPR Behavior

With a properly designed and characterized thermal solution, it is anticipated that FORCEPR# will only be asserted for very short periods of time when running the most power intensive applications. The processor performance impact due to these brief periods of TCC activation is expected to be so minor that it would be immeasurable. However, an under-designed thermal solution that is not able to prevent excessive assertion of FORCEPR# in the anticipated ambient environment may:

- Cause a noticeable performance loss.
- Result in prolonged operation at or above the specified maximum operating temperature and affect the long-term reliability of the processor.
- May be incapable of cooling the processor even when the TCC is active continuously (in extreme situations).

12.1.2.8 Low-Power States and FORCEPR Behavior

Depending on package power levels during package C-states, outbound FORCEPR# may de-assert while the processor is idle as power is removed from the signal. Upon wake up, if the processor is still hot, the FORCEPR# will re-assert, although typically package idle state residency should resolve any thermal issues. The PECI mechanism

is fully operational during all C-states and it is expected that the platform continues to manage processor IA core and package thermals even during idle states by regularly polling for thermal data over PECI.

12.1.2.9 THERMTRIP Signal

Regardless of enabling the automatic or on-demand modes, in the event of a catastrophic cooling failure, the package will automatically shut down when the silicon has reached an elevated temperature that risks physical damage to the product. At this point, the THERMTRIP# signal will go active.

12.1.2.10 Critical Temperature Detection

Critical Temperature detection is performed by monitoring the package temperature. This feature is intended for graceful shutdown before the THERMTRIP# is activated. However, the processor execution is not guaranteed between critical temperature and THERMTRIP#. If the Adaptive Thermal Monitor is triggered and the temperature remains high, a critical temperature status and sticky bit are latched in the PACKAGE_THERM_STATUS (1B1h) MSR and the condition also generates a thermal interrupt, if enabled. For more details on the interrupt mechanism, refer to *Intel® 64 Architectures Software Developer's Manual* (671200) .

12.1.3 Assured Power

Assured Power form a design option where the processor's behavior and package Processor Base Power are dynamically adjusted to a desired system performance and power envelope. Assured Power technologies offer opportunities to differentiate system design while running active workloads on select processor SKUs through scalability, configuration and adaptability. The scenarios or methods by which each technology is used are customizable but typically involve changes to PL1 and associated frequencies for the scenario with a resultant change in performance depending on system's usage. Either technology can be triggered by (but are not limited to) changes in OS power policies or hardware events such as docking a system, flipping a switch or pressing a button. cTDP and LPM are designed to be configured dynamically and do not require an operating system reboot.

NOTES

- FORCEPR# events should be triggered after BIOS active. Triggering FORCEPR after BIOS is active should be ensured as it is essential for system stability.
- Assured Power technologies are not battery life improvement technologies.

12.1.3.1 Assured Power Modes

NOTE

Assured Power availability may vary between the different SKUs.

With cTDP, the processor is now capable of altering the maximum sustained power with an alternate processor IA core base frequency. Assured Power allows operation in situations where extra cooling is available or situations where a cooler and quieter mode of operation is desired.

cTDP consists of three modes as shown in the following table.

Table 37. Assured Power (cTDP)

SOC Power Characteristic	Description of Characteristic	SOC Design Considerations
Maximum Turbo Power	The maximum sustained (>1s) power dissipation of the processor as limited by current and/or temperature controls. Instantaneous power may exceed Maximum Turbo Power for short durations (<=10ms). Maximum Turbo Power is configurable by system vendor and can be system specific.	Intel performance advocacy for power delivery and transient thermal solution design (PL2)
Base Power	The time-averaged power dissipation that the processor is validated to not exceed during manufacturing while executing an Intel-specified high complexity workload at Base Frequency and at the operating temperature as specified in the Datasheet for the SKU segment and configuration.	Intel reference performance advocacy for sustained thermal solution design (PL1)
Minimum Assured Power	Min Assured Power is a performance advocacy determined by running a complex scenario defined by Intel. Every product SKU stack has guidance on Min Assured Power for thermal chassis design. Represents Intel specified min PL1 that needs to be taken for thermal design to get the advocated performance experience. Min Assured Power is the performance vs power cross-over point across the product SKU stack.	Intel minimum performance advocacy for sustained thermal solution design (PL1)
High Concurrency Power	High Concurrency Power is a functional characteristic determined by running a complex scenario defined by Intel. Every SoC consumes a minimum power during a max connected case. This is a functional characteristic, not intended to indicate performance floor. Scenario takes into consideration IO ports, compute IPs concurrency (CPU, GPU, IPU) along with memory BW. Temperature assumption of spec limit. Manufacturing screening is done to exclude parts that don't meet the target power for the high concurrency scenario. The SoC may not honor PL1 values set lower than high concurrency power during the high concurrency scenario.	Thermal solution capability required to support the high concurrency scenario as described

In each mode, the Intel® Turbo Boost Technology 2.0 power limits are reprogrammed along with a new OS controlled frequency range. The Intel Dynamic Tuning driver assists in Processor Base Power operation by adjusting processor PL1 dynamically. The cTDP mode does not change the maximum per-processor IA core turbo frequency.

12.1.3.2 Low Power Mode

Low-Power Mode (LPM) can provide cooler and quieter system operation. By combining several active power limiting techniques, the processor can consume less power while running at equivalent low frequencies. Active power is defined as processor power consumed while a workload is running and does not refer to the power consumed during idle modes of operation. LPM is only available using the Intel® Dynamic Tuning (Intel® DTT/Intel® DPTF) driver.

Through the Intel® Dynamic Tuning (Intel® DTT/Intel® DPTF) driver, LPM can be configured to use each of the following methods to reduce active power:

- Restricting package power control limits and Intel® Turbo Boost Technology availability
- Off-Lining processor IA core activity (Move processor traffic to a subset of cores)
- Placing a processor IA Core at LFM or LSF (Lowest Supported Frequency)
- Utilizing IA clock modulation
- LPM power as listed in the Processor Base Power Thermal and Power Specifications on page 122 table is defined at a point which processor IA core working at LSF, GT = RPn and 1 IA core active

Off-lining processor IA core activity is the ability to dynamically scale a workload to a limited subset of cores in conjunction with a lower turbo power limit. It is one of the main vectors available to reduce active power. However, not all processor activity is ensured to be able to shift to a subset of cores. Shifting a workload to a limited subset of cores allows other processor IA cores to remain idle and save power. Therefore, when LPM is enabled, less power is consumed at equivalent frequencies.

Minimum Frequency Mode (MFM) of operation, which is the Lowest Supported Frequency (LSF) at the LFM voltage, has been made available for use under LPM for further reduction in active power beyond LFM capability to enable cooler and quieter modes of operation.

12.1.4 Intel Memory Thermal Management

DRAM Thermal Aggregation

P-Unit firmware is responsible for aggregating DRAM temperature sources into a per-DIMM reading as well as an aggregated virtual 'max' sensor reading. At reset, MRC communicates to the MC the valid channels and ranks as well as DRAM type. At that time, Punit firmware sets up a valid channel and rank mask that is then used in the thermal aggregation algorithm to produce a single maximum temperature.

DRAM Thermal Monitoring

- DRAM thermal sensing Periodic DDR thermal reads from DDR
- DRAM thermal calculation Punit reads of DDR thermal information direct from the memory controller (MR4 or MPR) Punit estimation of a virtual maximum DRAM temperature based on per-rank readings. Application of thermal filter to the virtual maximum temperature.

DRAM Refresh Rate Control

The MRC will natively interface with MR4 or MPR readings to adjust DRAM refresh rate as needed to maintain data integrity. This capability is enabled by default and occurs automatically. Direct override of this capability is available for debug purposes, but this cannot be adjusted during runtime.

DRAM Bandwidth Throttling (Change to DDR Bandwidth Throttling)

Control for bandwidth throttling is available through the memory controller. Software may program a percentage bandwidth target at the current operating frequency and that used to throttle read and write commands based on the maximum memory MPR/MR4 reading.

12.2 Processor Base Power Thermal and Power Specifications

Table 38. General Notes

Note	Definition
1	The Processor Base Power and Assured Power (cTDP) values are the average power dissipation in operating temperature condition limit, for the SKU Segment and Configuration, for which the processor is validated during manufacturing when executing an associated Intel-specified high-complexity workload at the processor IA core frequency corresponding to the configuration and SKU.
2	Processor Base Power workload may consist of a combination of processor IA core intensive and graphics core intensive applications.
3	Can be modified at runtime by MSR writes, with MMIO and with PECI commands.
4	'Turbo Time Parameter' is a mathematical parameter (units of seconds) that controls the processor turbo algorithm using a moving average of energy usage. Do not set the Turbo Time Parameter to a value less than 0.1 seconds. refer to Platform Power Control on page 112 for further information.
5	The shown limit is a time averaged-power, based upon the Turbo Time Parameter. Absolute product power may exceed the set limits for short durations or under virus or uncharacterized workloads.
6	The Processor will be controlled to a specified power limit. If the power value and/or 'Turbo Time Parameter' is changed during runtime, it may take a short time (approximately 3 to 5 times the 'Turbo Time Parameter') for the algorithm to settle at the new control limits.
7	This is a hardware default setting and not a behavioral characteristic of the part.
8	For controllable turbo workloads, the PL2 limit may be exceeded for up to 10ms.
9	LPM power level is an opportunistic power and is not a guaranteed value as usages and implementations may vary.
10	Power limits may vary depending on if the product supports the Minimum Assured Power (cTDP Down) and/or Maximum Assured Power (cTDP Up) modes. Default power limits can be found in the PKG_PWR_SKU MSR (614h).
11	The processor tile does not reach maximum sustained power simultaneously since the sum of all active circuit's estimated power budget is controlled to be equal to or less than the specified PL1 limit.
12	Minimum Assured Power(cTDP Down) is based on 128EU equivalent graphics configuration. Minimum Assured Power(cTDP Down) does not decrease the number of active Processor Graphics EUs but relies on Power Budget Management (PL1) to achieve the specified power level.
13	May vary based on SKU.
14	• The formula of PL2=PL1*1.25 is the hardware. • PL2- Processor opportunistic higher Average Power with limited duration controlled by Tau_PL1 setting, the larger the Tau, the longer the PL2 duration.

continued...

Note	Definition
15	Processor Base Power workload does not reflect various I/O connectivity cases such as Thunderbolt.
16	Hardware default of PL1 Tau=1s. By including the benefits available from power and thermal management features the recommended is to use PL1 Tau=28s.
17	PL1 Tau max recommendation value is the default value in the BIOS/BKC and this value is being tested.

Table 39. Processor Base Power Specifications (Processor)

Processor	Native GPU Configuration	CPU Configuration	Configuration			Processor P/E Core Frequency [GHz]	Processor Base Power [W]	General Notes
6C	2Xe	2P+0E+4LP_E	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	2.5 GHz	28	1,9,10
					LP E-Core	N/A		
				Processor Base power	P-Core	1.5 GHz	15	
					LP E-Core	1.4 GHz		
				Minimum Assured Power (cTDP Down)	P-Core	1.0 GHz	10	
					LP E-Core	N/A		
				Low Frequency Mode - LFM		0.4 GHz	N/A	
			Graphics Core Frequency	Graphics Frequency		0.4 GHz	N/A	
				Low Frequency Mode - LFM		0.1 GHz		
5C	2Xe	1P+0E+4LP_E	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	2.5 GHz	28	1,9,10
					LP E-Core	N/A		
				Processor Base power	P-Core	1.5 GHz	15	
					LP E-Core	1.4 GHz		
				Minimum Assured Power (cTDP Down)	P-Core	1.0 GHz	10	
					LP E-Core	N/A		
				Low Frequency Mode - LFM		0.4 GHz	N/A	
			Graphics Core Frequency	Graphics Frequency		0.4 GHz	N/A	
				Low Frequency Mode - LFM		0.1 GHz		

12.3 Thermal and Power Specifications

Table 40. Package Turbo Specifications Intel® Core™ Processor (Series 3) Processor)

Processor	Native GPU Configuration	Processor IA Cores, Graphics, Configuration and Processor Base Power	Parameter	Minimum	Tau MSR Max Value	Recommended Value	Units	General Notes
6C	2Xe	2P+0E +4LP_E	Power Limit 1 Time (PL1 Tau)	0.1	448	28	S	3,4,5,6,7,8,14,16,17
			Power Limit 1 (PL1)	N/A	N/A	15	W	

continued...

Processor	Native GPU Configuration	Processor IA Cores, Graphics, Configuration and Processor Base Power	Parameter	Minimum	Tau MSR Max Value	Recommended Value	Units	General Notes
			Power Limit 2 (PL2)	N/A	N/A	35	W	
5C	2Xe	1P+0E +4LP_E	Power Limit 1 Time (PL1 Tau)	0.1	448	28	S	3,4,5,6,7,8,14,16,17
			Power Limit 1 (PL1)	N/A	N/A	15	W	
			Power Limit 2 (PL2)	N/A	N/A	30	W	

Notes:
- No Specifications for Min/Max PL1/PL2 values.
- Hardware default of PL1 Tau=1s, By including the benefits available from power and thermal management features the recommended is to use PL1 Tau=28s for less than 45W .
- PL2- Processor opportunistic higher Average Power – Reactive, Limited Duration controlled by Tau_PL1 setting.
  PL1 Tau - PL1 average power is controlled via PID algorithm with this Tau, The larger the Tau, the longer the PL2 duration.
- System cooling solution and designs found to not being able to support the Performance TauPL1, adjust the TauPL1 to cooling capability.

Table 41. Operating Temperature Specifications Intel® Core™ Processor (Series 3) Processor)

Segment	Package Turbo Parameter	Temperature Range		Processor Base Power Specification Temperature Range		Units	Notes
		Minimum	Maximum	Minimum	Maximum		
All	Operating temperature	0	100	35	100	ºC	1,2

Notes:
1. The thermal solution needs to ensure that the processor temperature does not exceed the Processor temperature range.
2. The processor operating temperature is monitored by Digital Temperature Sensors (DTS). For DTS accuracy, refer to Digital Thermal Sensor on page 116

12.4 Error and Thermal Protection Signals

Table 42. Error and Thermal Protection Signals

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
CATERR#	**Catastrophic Error:** This signal indicates that the system has experienced a catastrophic error and cannot continue to operate. The processor will set this signal for non-recoverable machine check errors or other unrecoverable internal errors. CATERR# is used for signaling the following types	O	OD	SE	All Processor Series

continued...

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
	of errors: Legacy MCERRs, CATERR# is asserted for 16 BCLKs. Legacy IERRs, CATERR# remains asserted until warm or cold reset.				
PECI	**Platform Environment Control Interface:** Supported over eSPI only. It is used primarily for thermal, power, and error management.	I/O	eSPI	SE	All Processor Series
FORCEPR#	**Processor Hot:** FORCEPR# goes active when the processor temperature monitoring sensor(s) detects that the processor has reached its maximum safe operating temperature. This indicates that the processor Thermal Control Circuit (TCC) has been activated, if enabled.	I	I:GTL	SE	All Processor Series
THERMTRIP#	**Thermal Trip:** The processor protects itself from catastrophic overheating by use of an internal thermal sensor. This sensor is set well above the normal operating temperature to ensure that there are no false trips. The processor will stop all executions when the operating temperature exceeds approximately 125 °C. This is signaled to the system by the THERMTRIP# pin.	O	OD	SE	All Processor Series

12.5 Thermal Sensor

The processor incorporates on-die Digital Thermal Sensors for thermal management.

12.5.1 Modes of Operation

The thermal sensors have two usages when enabled:

1. One use is to provide the temperature of the processor in units of 1 ºC.
2. The second use is to allow programmed trip points to cause alerts to SW or in the extreme case shutdown. Temperature may be provided without having any SW alerts set.

There are two thermal alert capabilities. One is for the catastrophic event (thermal runaway) which results in an immediate system power down. The other alert provides an indication to the platform that a particular temperature has been caused. This second alert needs to be routed to SMI or SCI based on SW programming.

12.5.2 Temperature Trip Point

The internal thermal sensors reports three trip points: Cool, Hot, and Catastrophic trip points in the order of increasing temperature.

Crossing the cool trip point when going from higher to lower temperature may generate an interrupt. Crossing the hot trip point going from lower to higher temp may generate an interrupt. Each trip point has control register bits to select what type of interrupt is generated.

Crossing the cool trip point while going from low to higher temperature or crossing the hot trip point while going from high to lower temperature will not cause an interrupt.

When triggered, the catastrophic trip point will transition the system to S5 unconditionally.

12.5.3 Thermal Reporting to EC

To support a platform EC that is managing the system thermals, the processor provides the ability for the EC to read the processor temperature over SMBus and/or over eSPI. If enabled, Power Management will drive the temperature directly to the SMBus and eSPI units. The EC will issue an SMBus read or eSPI OOB channel request and receives a single byte of data. The EC must be connected to either SMLink1 or eSPI for thermal reporting support.

12.5.4 Thermal Trip Signal

The processor provides SOCHOT# signal to indicate that it has exceeded some temperature limit. The limit is set by BIOS. The temperature limit is compared to the present temperature. If the present temperature is greater than the programmed value then the pin is asserted.

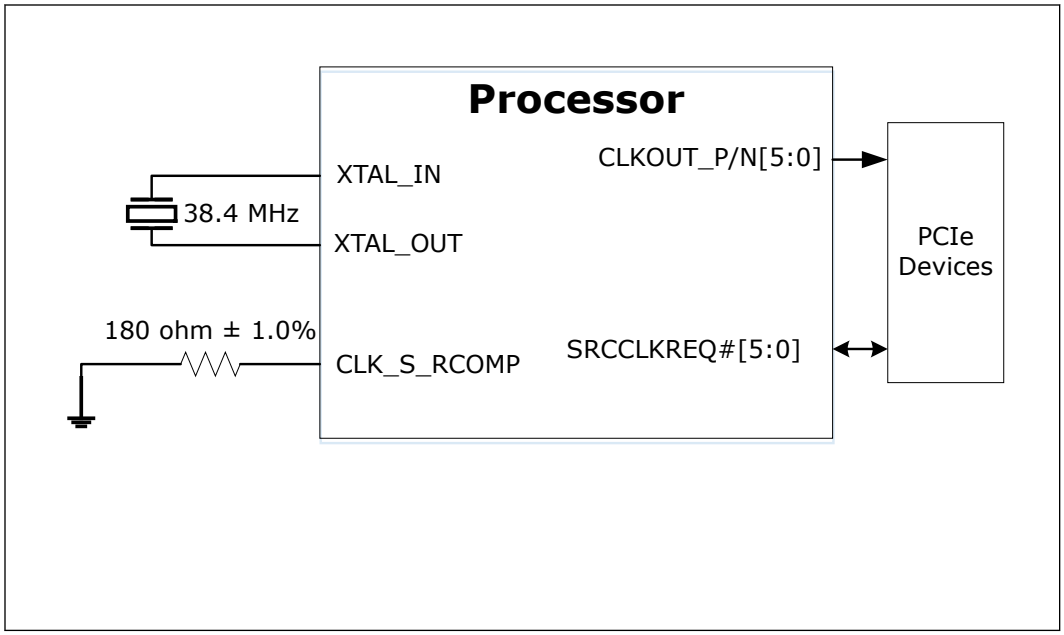
SOCHOT# is an O/D output and requires a Pull-up on the motherboard.

The processor evaluates the temperature from the thermal sensor against the programmed temperature limit every 1 second.

13.0 System Clocks

13.1 ICC

Figure 10. ICC Diagram



13.1.1 Signal Description

Table 43. Signal Description

Signal Name	Type	SSC Capable	Description
CLKOUT_P0 CLKOUT_P1 CLKOUT_P2 CLKOUT_P3 CLKOUT_P4 CLKOUT_P5 CLKOUT_N0 CLKOUT_N1 CLKOUT_N2 CLKOUT_N3 CLKOUT_N4	O	Yes	PCI Express* Clock Output: Serial Reference 100 MHz PCIe* specification compliant differential output clocks to PCIe devices. CLKOUT_P/N[5:0]= Can be used for PCIe Gen1, Gen2, Gen3, and Gen4 support.

continued...

Signal Name	Type	SSC Capable	Description
CLKOUT_N5			
GPP_C09/**SRCCLKREQ0#** GPP_C10/**SRCCLKREQ1#** GPP_C11/**SRCCLKREQ2#** GPP_C12/**SRCCLKREQ3#** GPP_C13/**SRCCLKREQ4#** GPP_C14/**SRCCLKREQ5#** GPP_D21/**UFS_REFCLK**	IOD		Clock Request: Serial Reference Clock request signals for PCIe* 100 MHz differential clocks The SRCCLKREQ*# signals can be configured to map to any of the PCD PCI Express* Root Ports while using any of the PCD CLKOUT differential pairs.
XTAL_IN	I		Crystal Input: Input connection for 38.4 MHz crystal to Processor.
XTAL_OUT	O		Crystal Output: Output connection for 38.4 MHz crystal to Processor.
CLK_S_RCOMP	Analog		Differential Clock Bias Reference: Used to set BIAS reference for differential clocks

Notes: 1. SSC = Spread Spectrum Clocking
2. The SRCCLKREQ# signals can be configured to map to any of the PCI Express Root Ports while using any of the clock output differential pairs.
3. Above consideration is not applicable when designing platform that does not use common motherboard concept.

13.2 IO Signal Pin States

Table 44. I/O Signal Pin States

Signal Name	Power Plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CLKOUT_P[0:5] CLKOUT_N[0:5]	Primary	Toggling	Toggling	OFF (Gated Low)
SRCCLKREQ[0:5]#	Primary	Un-driven	Un-driven	Un-driven

1. Reset reference for primary well pins is RSMRST#.

14.0 Real Time Clock (RTC)

The Processor contains a real-time clock functionally compatible with the Motorola* MC146818B. The real-time clock has 256 bytes of battery-backed RAM.

The real-time clock performs two key functions:

- Keep track of the time of day.
- Store system data even when the system is powered down as long as the RTC power well is powered.

The RTC operates on a 32.768 kHz oscillating source and a 1.5 V battery or system battery if configured by design as the source.

The RTC also supports two lockable memory ranges. By setting bits in the configuration space, two 8-byte ranges can be locked to read and write accesses. This prevents unauthorized reading of passwords or other system security information.

The RTC also supports a date alarm that allows for scheduling a wake-up event up to month in advance.

Table 45. Acronyms

Acronyms	Description
BCD	Binary Coded Decimal
CMOS	Complementary Metal Oxide Semiconductor. A manufacturing process used to produce electronics circuits, but in reference to RTC is used interchangeably as the RTC's RAM i.e. clearing CMOS meaning to clear RTC RAM.
ESR	Equivalent Series Resistance. Resistive element in a circuit such as a clock crystal.
GPI	General Purpose Input
PPM	Parts Per Million. Used to provide crystal accuracy or as a frequency variation indicator.
RAM	Random Access Memory

14.1 Signal Description

Signal Name	Type	Description
RTCX1	I	**Crystal Input 1:** This signal is connected to the 32.768 kHz crystal (max 50 kohm ESR). If no external crystal is used, then RTCX1 can be driven with the desired clock rate. Maximum voltage allowed on this pin is 1.5 V.
RTCX2	O	**Crystal Input 2:** This signal is connected to the 32.768 kHz crystal (max 50 kohm ESR). If no external crystal is used, then RTCX2 must be left floating.
RTCRST#	I	**RTC Reset**: When asserted, this signal resets register bits in the RTC well.

continued...

Signal Name	Type	Description
		Note: 1. Unless CMOS is being cleared (only to be done in the G3 power state) with a jumper, the RTCRST# input must always be high when all other RTC power planes are on.
SRTCRST#	I	**Secondary RTC Reset**: This signal resets the manageability register bits in the RTC well when the RTC battery is removed. *Notes:* 1. The SRTCRST# input must always be high when all other RTC power planes are on. 2. SRTCRST# and RTCRST# should not be shorted together.

14.2 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
RTCRST#	RTC	HIGH	HIGH	HIGH
SRTCRST#	RTC	HIGH	HIGH	HIGH
Note: 1. Reset reference for RTC well pin is RTCRST#.				

15.0 Memory

15.1 System Memory Interface

15.1.1 Processor SKU Support Matrix

Table 46. DDR Support Matrix Table

Technology	DDR5 [5]	LPDDR5x [7]
Processor	6C	6C
Maximum Frequency [MT/s]	6400 [4]	Type 3: 1R/2R - 7466[4]
Channels	2 x32 Channels	4 x16 Channels
SPC [1]	1	-
Maximum RPC [2]	2	2
Die Density [Gb]	16, 24, 32	16,12, 24
VDDQ [V]	1.1 [5]	0.52 [6]
VDD2 [V]	1.1 [5]	1.065 [6]
Ballmap Mode	NIL	NIL

Notes:
1. 1SPC refers to one DIMM Slot per x64 Channel.
2. RPC = Rank Per Channel
3. Memory down of all technologies should be implemented homogeneous means that all DRAM devices should be from the same vendor and have the same part number. Implementing a mix of DRAM devices may cause serious signal integrity and functional issues.
4. POR top speed refers to Processor top SKU. Other SKUs may use lower memory speed, refer to ark.intel.com for top memory speed. .
5. DDR5 DIMM PMIC input voltage is 5V, DRAM input voltage is 1.1V.
6. LPDDR5x: VDD2 is processor and DRAM voltage , VDDQ is processor and DRAM voltage. .
7. LPDDR5x technology supports BG Mode and 16 Bank Mode, according to JEDEC spec. The processor supports BG Mode and 16 Bank Mode. Bank Mode may vary according to SAGV Point.
8. Pending DRAM samples availability and eco system readiness.

Table 47. DDR Technology Support Matrix

Technology	Form Factor	Ball Count	Processor
DDR5	**SoDIMM/CSODIMM**	262	All
DDR5	**x8 SDP (1R)**[1]	78	All
DDR5	**x16 SDP (1R)**[1]	106	All
LPDDR5x	**x64 (1R, 2R)**[1]	496	All
LPDDR5x	**x32 (1R, 2R)**[1]	315	All

NOTE

Memory down of all technologies should be implemented homogeneously, which means that all DRAM devices should be from the same vendor and have the same part number. Implementing a mix of DRAM devices may cause serious signal integrity and functional issues, (all DRAMs in the system must be from same Part Number).

15.1.2 Supported Memory Modules and Devices

Table 48. Supported DDR5 Non-ECC SoDIMM/CSoDIMM Module Configurations

Raw Card Version	DIMM Capacity [GB]	DRAM Device Technology [Gb]	DRAM Organization	# of DRAM Devices	# of Ranks	# of Row/Col Address Bits	# of Banks Inside DRAM	Page Size [K]
A , F	16	16	2048M x 8	8	1	17/10	16	8
C , H	8	16	1024M x 16	4	1	17/10	8	8
B , G	32	16	2048M x 8	16	2	17/10	16	8
A , F	24	24	3072M x 8	8	1	17/10	32	8
C , H	12	24	1536M x 16	4	1	17/10	16	8
B , G	48	24	3072M x 8	16	2	17/10	32	8
A , F	32	32	8182M x 8	8	1	18/10	64	8
C , H	16	32	2048M x 16	4	1	18/10	32	8
B , G	64	32	8182M x 8	16	2	18/10	64	8

Table 49. Supported DDR5 Memory Down Device Configurations

Maximum System Capacity [GB][2]	PKG Type (Die bits x Package bits)	DRAM Organization / Package Type	Package Density [Gb]	Die Density [Gb]	Dies Per Channel	Rank Per Channel	PKGs Per channel	Physical Device Rank	Banks Inside DRAM	Page Size [K]
16	SDP 8x8	2048M x 8	16	16	8	1	8	1	16	8
8	SDP 16x16	1024M x 16	16	16	4	1	4	1	8	8
24	SDP 8x8	3072M x 8	24	24	8	1	8	1	32	8
12	SDP 16x16	1536M x 16	24	24	4	1	4	1	16	8
32	SDP 8x8	3072M x 8	32	32	8	1	8	1	64	8
16	SDP 16x16	1536M x 16	32	32	4	1	4	1	32	8

Notes: 1. For SDP: 1Rx16 using 16 GB die density - the maximum system capacity is 16 GB
2. Maximum system capacity, refer to system populated with same memory down devises

Table 50. Supported LPDDR5/x x32 DRAMs Configurations

Maximum System Capacity [GB][3]	PKG Type [2]	(Die bits per Ch x PKG bits)	Die Density [Gb]	PKG Density [Gb]	Rank Per PKGs
6	DDP	16x32	12	24	1
12	QDP	16x32	12	48	2

continued...

Maximum System Capacity [GB][3]	PKG Type [2]	(Die bits per Ch x PKG bits)	Die Density [Gb]	PKG Density [Gb]	Rank Per PKGs
24	ODP	16x32	12	96	2
8	DDP	16x32	16	32	1
16	QDP	16x32	16	64	2
32	ODP	16x32	16	128	2
12	DDP	16x32	24	48	1
24	QDP	16x32	24	96	2
48	ODP	16x32	24	192	2

Notes: 1. x32 BGA devices are 315 balls
2. DDP - Dual Die Package, QDP - Quad Die Package, ODP - Octal Die Package
3. Maximum system capacity refers to system with all 4 sub-channels populated

Table 51. Supported LPDDR5/x x64 DRAMs Configurations

Maximum System Capacity [GB][2]	PKG Type	(Die bits per Ch x PKG bits)[2]	Die Density [Gb]	PKG Density [Gb]	DRAM Channels Per PKGs	Rank Per PKGs
6[1]	QDP	16x64	12	48	4	1
12[1]	ODP	16x64	12	96	4	2
8[1]	QDP	16x64	16	64	4	1
16[1]	ODP	16x64	16	128	4	2
12[1]	QDP	16x64	24	96	4	1
24[1]	ODP	16x64	24	192	4	2

Notes: 1. QDP = Quad Die Package, ODP-Octal Die Package
2. Maximum system capacity refers to system with all 4sub-channels populated

15.1.3 System Memory Timing Support

The IMC supports the following DDR Speed Bin, CAS Write Latency (CWL), and command signal mode timings on the main memory interface:

- tCL = CAS Latency
- tRCD = Activate Command to READ or WRITE Command delay
- tRP = PRECHARGE Command Period
- tRPb = per-bank PRECHARGE time
- tRPab = all-bank PRECHARGE time
- CWL = CAS Write Latency
- Command Signal modes:
  - — 2N indicates a new DDR5/LPDDR5/x command may be issued every 2 clocks

Table 52. DDR5 System Memory Timing Support

DRAM Device	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRP (ns)	CWL (tCK)	CMD Mode
DDR5	4800	40	16.00	16.00	38	2N
DDR5	5600	46	16.00	16.00	44	2N
DDR5	6000	48	16.00	16.00	46	2N
DDR5	6400	52	16.00	16.00	50	2N

Table 53. LPDDR5/x System Memory Timing Support

DRAM Device	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRPpb (ns)	tRPab (ns)	WL (tCK) Set B
LPDDR5	6400	17	18	18	21	16
LPDDR5x	6800	18	18	18	21	17

15.1.3.1 SAGV Points

Refer to Intel System Agent Enhanced SpeedStep ® Technology on page 98 for more details

Table 54. SA Speed Enhanced Speed Steps (SA-GV) and Gear Mode Frequencies

Processor	Technology	Rank Config	DDR Maximum Rate [MT/s]	SAGV-LowBW	SAGV-MedBW	SAGV-HighBW	SAGV- High Performance
all	LPDDR5x Type 3	1R/2R	6800	2400 G4	4800 G4	6000 G4	6800 G4
	DDR5	1R/2R	6400	3200 G4	4800 G4	6000 G4	6400 G4

Notes: 1. Wildcat Lake supports dynamic gearing technology where the Memory Controller can run at 1:2 (Gear-2 mode) or 1:4 (Gear-4 mode) ratio of DRAM speed. The gear ratio is the ratio of DRAM speed to Memory Controller Clock .

MC Channel Width equal to DDR Channel width multiply by Gear Ratio.

2. SA-GV modes:
   a. **LowBW-** Low frequency point, Minimum Power point. Characterized by low power, low BW, high latency. The system will stay at this point during low to moderate BW consumption.
   b. **MedBW -** Tuned for balance between power & performance.
   c. **HighBW -** Characterized by high power, low latency, moderate BW also used as RFI mitigation point.
   d. **MaxBW/Lowest latency** Lowest Latency point, peak BW and highest power.

DDR Frequency Shifting

DDR interfaces emit electromagnetic radiation which can couple to the antennas of various radios that are integrated in the system, and cause radio frequency interference (RFI). The DDR Radio Frequency Interference Mitigation (DDR RFIM) feature is primarily aimed at resolving narrowband RFI from DDR5 andLPDDR5/x technologies for the Wi-Fi* high and ultra-high bands (~5-7 GHz) . By changing the DDR data rate, the harmonics of the clock can be shifted out of a radio band of interest, thus mitigating RFI to that radio. This feature is working with SAGV on, the 3rd SAGV point is used as RFI mitigation point

15.1.4 Memory Controller (MC)

The integrated memory controller is responsible for transferring data between the processor and the DRAM as well as the DRAM maintenance. There is one instances of MC, the controller is capable of supporting up to four channels of LPDDR5/x, two channels of DDR5.

15.1.5 System Memory Frequency

In all modes, the frequency of system memory is the lowest frequency and highest latency of all memory modules placed in the system, as determined through the SPD registers on the memory modules.

15.1.6 Technology Enhancements of Intel® Fast Memory Access (Intel® FMA)

The following sections describe the Just-in-Time Scheduling, Command Overlap, and Out-of-Order Scheduling Intel® FMA technology enhancements.

Just-in-Time Command Scheduling

The memory controller has an advanced command scheduler where all pending requests are examined simultaneously to determine the most efficient request to be issued next. The most efficient request is picked from all pending requests and issued to system memory Just-in-Time to make optimal use of Command Overlapping. Thus, instead of having all memory access requests go individually through an arbitration mechanism forcing requests to be executed one at a time, they can be started without interfering with the current request allowing for concurrent issuing of requests. This allows for optimized bandwidth and reduced latency while maintaining appropriate command spacing to meet system memory protocol.

Command Overlap

Command Overlap allows the insertion of the DRAM commands between the Activate, Pre-charge, and Read/Write commands normally used, as long as the inserted commands do not affect the currently executing command. Multiple commands can be issued in an overlapping manner, increasing the efficiency of system memory protocol.

Out-of-Order Scheduling

While leveraging the Just-in-Time Scheduling and Command Overlap enhancements, the IMC continuously monitors pending requests to system memory for the best use of bandwidth and reduction of latency. If there are multiple requests to the same open page, these requests would be launched in a back to back manner to make optimum use of the open memory page. This ability to reorder requests on the fly allows the IMC to further reduce latency and increase bandwidth efficiency.

15.1.7 Data Scrambling

The system memory controller incorporates a Data Scrambling feature to minimize the impact of excessive di/dt on the platform system memory VRs due to successive 1s and 0s on the data bus. Past experience has demonstrated that traffic on the data bus is not random and can have energy concentrated at specific spectral harmonics

creating high di/dt which is generally limited by data patterns that excite resonance between the package inductance and on die capacitances. As a result, the system memory controller uses a data scrambling feature to create pseudo-random patterns on the system memory data bus to reduce the impact of any excessive di/dt.

15.1.8 Data Swapping

By default, the processor supports on-board data swapping in two manners (for all segments and DRAM technologies):

- Bit swapping is allowed within each Byte for all DDR technologies.
- LPDDR5/x: x16 sub-channels can be swizzled within their x64 MC.
- LPDDR5/x: Byte swapping is allowed within each x16 Channel.
- DDR5 x32 sub-channels can be swizzle within their x64 MC.
- DDR5: Byte swapping is allowed within each x32 Channel.

15.1.9 LPDDR5x CMDADD Ascending and Descending

LPDDR5/x support Ascending / descending that swap CA and CS signals connectivity order.

Table 55. LPDDR5/x CMD/ADD Ascending and Descending

Ascending	Descending
CA6	CA0
CA5	CA1
CA4	CS_1
CA3	CS_0
CA2	CA2
CS_0	CA3
CS_1	CA4
CA1	CA5
CA0	CA6

NOTE

Ascending / descending can be performed in every x16 sub channel.

15.1.10 DDR IO Interleaving

The processor supports I/O interleaving, which has the ability to swap DDR bytes for routing considerations BIOS. configures the I/O interleaving mode before DDR

15.1.11 DRAM Clock Generation

Each support rank has a differential clock pair for DDR5. Each sub-channel has a (CK_P/N and WCK_P/N) differential clock pair for LPDDR5/x.

15.1.12 DRAM Reference Voltage Generation

Read Vref is generated by the memory controller in all technologies. Write Vref is generated by the DRAM in all technologies. Command Vref is generated by the DRAM in LPDDR5/x. In all cases, it has small step sizes and is trained by MRC.

15.1.13 Data Swizzling

All Processor Lines have no die-to-package DDR swizzling.

15.1.14 Error Correction With Standard RAM

In-Band error-correcting code (IBECC) correct single-bit memory errors in standard, non-ECC memory.

Supported only in Chrome systems with memory channels symmetrical population (both channels must be populated with same memory size/ranks/dram type).

Processor performance might be lower when IBECC enabled due to memory bandwidth consumed by IBECC.

NPU performance may be lower when IBECC and TME/MK-TME are enabled.

15.1.15 Post Package Repair (PPR)

PPR is supported according to JEDEC Spec.

BIOS can identify a single Row failure per Bank in DRAM and perform Post Package Repair (PPR) to exchange failing Row with spare Row.

PPR can be supported only with DRAM that supports PPR according to Jedec spec.

15.1.16 RFM

RFM is supported according to JEDEC spec.

15.1.17 In-Memory Analytics Accelerator

Intel® In-Memory Analytics Accelerator (Intel® IAA) is a hardware accelerator that provides very high throughput compression and decompression along with encryption and decryption.

The Accelerator allows storing columnar databases in compressed form, decreasing memory footprint. In addition to increased effective memory capacity, this also reduces memory bandwidth by performing the filter function used for database queries on the fly, thereby avoiding the use of memory bandwidth for uncompressed raw data transfer.

15.2 Integrated Memory Controller (IMC) Power Management

The main memory is power managed during normal operation and in low-power ACPI C-states.

15.2.1 DRAM Power Management and Initialization

The processor implements extensive support for power management on the memory interface.

The DRAM Powerdown is one of the power-saving means. When DRAM is in Powerdown state, the internal DDR clock is disabled and the DDR power is reduced. The power-saving differs according to the selected mode and the DDR type used. For more information, refer to the IDD table in the DDR specification.

The processor supports three different types of power-down modes in package C0 state. The different power-down modes can be enabled through configuring PM PDWN config register.

The different power-down modes supported are:

- **No power-down**:
- **Pre-charged Power-down (PPD):** This mode is entered if all banks in DDR are pre-charged when entering Powerdown state. Power-saving in this mode is intermediate. Power consumption is defined by IDD2P. Exiting this mode is defined by tXP. In this mode when waking-up, all page-buffers are empty.

The Powerdown state is determined per rank, whenever it is inactive. Each rank has an idle counter. The idle-counter starts counting as soon as the rank has no accesses, and if it expires, the rank may enter power-down while no new transactions to the rank arrive to queues. It is important to understand that since the power-down decision is per rank, the IMC can find many opportunities to power down ranks, even while running memory intensive applications; the savings are significant (may be few Watts, according to DDR specification). This is significant when each channel is populated with more ranks.

Selection of power modes should be according to power-performance or a thermal trade-off of a given system:

- When trying to achieve maximum performance and power or thermal consideration is not an issue: use no power-down
- In a system which tries to minimize power-consumption, try using the deepest power-down mode possible
- In high-performance systems with dense packaging (that is, tricky thermal design) the power-down mode should be considered in order to reduce the heating and avoid DDR throttling caused by the heating.

The idle timer expiration count defines the # of DCLKs that a rank is idle that causes entry to the selected power mode. As this timer is set to a shorter time the IMC will have more opportunities to put the DDR in power-down. There is no BIOS hook to set this register. Customers choosing to change the value of this register can do it by changing it in the BIOS. For experiments, this register can be modified in real time if BIOS does not lock the IMC registers.

15.2.1.1 Conditional Self-Refresh

During S0 idle state, system memory may be conditionally placed into self-refresh state when the processor is in package C0 or deeper power state. Refer to Intel® Rapid Memory Power Management (Intel® RMPM) on page 91 for more details on conditional self-refresh with Intel® HD Graphics enabled.

When entering the S0 conditional self-refresh, the processor IA core flushes pending cycles and then enters SDRAM ranks that are not used by the processor graphics into self-refresh.

The target behavior is to enter self-refresh for package C0 or deeper power states as long as there are no memory requests to service.

15.2.1.2 Dynamic Power-Down

Dynamic power-down of memory is employed during normal operation. Based on idle conditions, a given memory rank may be powered down. The IMC implements aggressive CKE control to dynamically put the DRAM devices in a power-down state. If dynamic power-down is enabled, all ranks are powered up before doing a refresh cycle and all ranks are powered down at the end of the refresh

15.2.1.3 DRAM IO Power Management

Unused signals should be disabled to save power and reduce electromagnetic interference. Clocks and CS signals are controlled per DIMM rank and will be powered down for unused ranks.

The I/O buffer for an unused signal should be tri-stated (output driver disabled), the input receiver (differential sense-amp) should be disabled. The input path should be gated to prevent spurious results due to noise on the unused signals (typically handled automatically when input receiver is disabled).

15.2.2 DDR Electrical Power Gating

The DDR I/O of the processor supports Electrical Power Gating (DDR-EPG) while the processor is at C3 or deeper power state.

In C3 or deeper power state, the processor internally gates VDDQ and VDD2 for the majority of the logic to reduce idle power while keeping all critical DDR pins such as CKE in the appropriate state.

In C8 or deeper power state, the processor internally gates VCCSA for all non-critical state to reduce idle power.

In C-state transitions, the DDR does not go through training mode and will restore the previous training information.

15.2.3 Power Training

BIOS MRC performing Power Training steps to reduce DDR I/O power while keeping reasonable operational margins still guaranteeing platform operation. The algorithms attempt to weaken ODT, driver strength and the related buffers parameters both on the MC and the DRAM side and find the best possible trade-off between the total I/O power and the operating margins using advanced mathematical models.

15.2.4 DVFS

Intel® Core™ Series 3 supports:

- **E-DVFSC:**mode intended to reduce the LPDDR5x RAM energy consumption. When E- DVFSC is enabled the memory Processor PHY interface shall operate with VDD2L 0.9 V rail at speed equal and below 3200 MT/s and switch to VDDH 1.065 V rail in speed above 3200 MT/s.

15.3 Signal Description

Table 56. DDR5 Memory Interface

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR0_DQ[3:0][7:0] DDR1_DQ[3:0][7:0]	**Data Buses**: Data signals interface to the SDRAM data buses. **Example:** DDR0_DQ2[5] refers to DDR channel 0, Byte 2, Bit 5.	I/O	DDR5	SE	All Processor Lines
DDR0_DQSP[3:0] DDR0_DQSN[3:0] DDR1_DQSP[3:0] DDR1_DQSN[3:0]	**Data Strobes:** Differential data strobe pairs. The data is captured at the crossing point of DQS during reading and write transactions. **Example:** DDR0_DQSP0 refers to DQSP of DDR channel 0, Byte 0.	I/O	DDR5	Diff	All Processor Lines
DDR0_CLK[1:0]_P DDR0_CLK[1:0]_N DDR1_CLK[1:0]_P DDR1_CLK[1:0]_N	**SDRAM Differential Clock:** Differential clocks signal pairs, pair per rank. The crossing of the positive edge and the negative edge of their complement are used to sample the command and control signals on the SDRAM.	O	DDR5	Diff	All Processor Lines
DDR0_CS[1:0] DDR1_CS[1:0]	**Chip Select**: (1 per rank). These signals are used to select particular SDRAM components during the active state. There is one Chip Select for each SDRAM rank. The Chip select signal is Active Low.	O	DDR5	SE	All Processor Lines
DDR0_CA[12:0] DDR1_CA[12:0]	**Command Address**: These signals are used to provide the multiplexed command and address to the SDRAM.	O	DDR5	SE	All Processor Lines
DDR_RCOMP		A	A	SE	All Processor Lines
DRAM_RESET#	O	CMOS	SE	All Processor Lines	

Table 57. LPDDR5/x Memory Interface

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR0_DQ[1:0][7:0] DDR1_DQ[1:0][7:0] DDR2_DQ[1:0][7:0] DDR3_DQ[1:0][7:0]	**Data Buses**: Data signals interface to the SDRAM data buses. **Example**: DDR0_DQ[1][5] refers to DDR channel 0, Byte 1, Bit 5.	I/O	LPDDR5/x	SE	All Processor Lines
DDR0_DQSP[1:0] DDR1_DQSP[1:0] DDR2_DQSP[1:0]	**Data Strobes**: Differential data strobe pairs. The data is captured at the crossing	I/O	LPDDR5/x	Diff	All Processor Lines

continued...

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR3_DQSP[1:0] DDR0_DQSN[1:0] DDR1_DQSN[1:0] DDR2_DQSN[1:0] DDR3_DQSN[1:0]	point of DQS during reading and write transactions.				
DDR0_CLK_N DDR0_CLK_P DDR1_CLK_N DDR1_CLK_P DDR2_CLK_N DDR2_CLK_P DDR3_CLK_N DDR3_CLK_P	**SDRAM Differential Clock**: Differential clocks signal pairs, pair per channel and package. The crossing of the positive edge and the negative edge of their complement are used to sample the command and control signals on the SDRAM.	O	LPDDR5/x	Diff	All Processor Lines
DDR0_CS[1:0] DDR1_CS[1:0] DDR2_CS[1:0] DDR3_CS[1:0]	**Chip Select**: (1 per rank). These signals are used to select particular SDRAM components during the active state. There is one Chip Select for each SDRAM rank. The Chip select signal is Active High.	O	LPDDR5/x	SE	All Processor Lines
DDR0_CA[6:0] DDR1_CA[6:0] DDR2_CA[6:0] DDR3_CA[6:0]	**Command Address**: These signals are used to provide the multiplexed command and address to the SDRAM.	O	LPDDR5/x	SE	All Processor Lines
DDR0_WCK[1:0]_P DDR0_WCK[1:0]_N DDR1_WCK[1:0]_P DDR1_WCK[1:0]_N DDR2_WCK[1:0]_P DDR2_WCK[1:0]_N DDR3_WCK[1:0]_P DDR3_WCK[1:0]_N	**Write Clocks**: WCK_N and WCK_P are differential clocks used for WRITE data capture and READ data output.	O	LPDDR5/x	Diff	All Processor Lines
DDR_RCOMP		A	A	SE	All Processor Lines
DRAM_RESET#	O	CMOS	SE	All Processor Lines	

16.0 USB Type-C Sub System

USB Type-C* is a cable and connector specification defined by USB-IF.

The USB Type-C sub-system supports USB 3.2, USB4, DPoC (DisplayPort over Type-C) protocols.

Thunderbolt™ 4 is a USB Type-C solution brand which requires the following elements:

- USB 2.0, USB 3.2 (2x10 Gb/s), USB 3.2/DP implemented at the connector.
- In additional, it requires USB4 implemented up to 40 Gbps, including Thunderbolt 3 compatibility as defined by USB4/USB-PD specs and 15 W of bus power
- Thunderbolt™ 4 solutions use (and prioritize) the USB4 PD entry mode (while still supporting Thunderbolt™ 3 alt mode)
- This product has the ability to support these requirements .

16.1 General Capabilities

- xHCI (USB 3.2 host controller) implemented in the processor.
- Intel® AMT/vPro over Thunderbolt™ docking.
- Support power saving when USB Type-C* disconnected.
- Support up to two simultaneous ports.
- DbC Enhancement for Low Power Debug until Pkg C6
- Host
  - Aggregate BW through the controller at least 3 GB/s, direct connection or over USB4.
  - Wake capable on each host port from S0i3, Sx.
- Device
  - D0i2 and D0i3 power gating
  - Wake capable on host initiated wakes when the system is in S0i3, Sx Available on all ports.
- Port Routing Control for Dual Role Capability
  - Needs to support SW/FW and ID pin based control to detect host versus device attach.
  - SW mode requires PD controller or other FW to control.
- USB-R device to host controller connection is over UTMI+ links.

Table 58. USB Type-C* Port Configuration

	Port	Supported Technologies
Group A	TCP 0	USB4[3]

continued...

	Port	Supported Technologies
	TCP 1	USB 3.2[2] DisplayPort[1]

Notes:
1. Display Port supported on Type-C Retimer Topology up to HBR3 link rate
2. USB 3.2 supported link rates:
   a. USB 3.2 Gen 1x1 (5 Gbps)
   b. USB 3.2 Gen 2x1 (10 Gbps)
   c. USB 3.2 Gen 2x2 (20 Gbps)
3. USB4 operating link rates (including both rounded and non-rounded modes for Thunderbolt™ 3 compatibility):
   a. USB4 Gen 2x2 (20 Gbps)
   b. USB4 Gen 3x2 (40 Gbps)
   c. 10.3125 Gbps, 20.625 Gbps per lane - Compatible to Thunderbolt™ 3 non-rounded modes.
4. USB 2.0 interface supported over Type-C connector.
5. Port group is defined as two ports sharing USB4 router, each router supports up to two display interfaces.

Table 59. USB Type-C* Lanes Configuration

Lane1	Lane2	Comments
USB4 / TBT3	USB4 / TBT3	Both lanes operate at same speed, one of (20.6g/10.3g/20g/10g)
USB4 / TBT3	No connect	20.6g/10.3g/20g/10g
No connect	USB4 / TBT3	
USB 3.2	USB 3.2	Multi-Lane USB 3.2 (Host Only), 2x10G = 20G
USB 3.2	No connect	Any combination of: USB 3.2 Gen 1x1 (5Gb/s) USB 3.2 Gen 2x1 (10Gb/s)
No connect	USB 3.2	
USB 3.2	DPx2	Any of HBR3/HBR2/HBR1/RBR, DisplayPort (v2.1 HBR3, DSC) and USB3.2 (10 Gbps)
DPx2	USB 3.2	
DPx4	Both lanes at same DP rate - no support for 2x DPx2 USB Type-C connector	Any of HBR3/HBR2/HBR1/RBR, DisplayPort (v2.1 HBR3, DSC)

Table 60. USB Type-C* Non-Supported Lane Configuration

Lane1	Lane2	Comments
-	PCIe* Gen3/2/1	No PCIe* native support
PCIe* Gen3/2/1	#	
-	USB4	No support for USB4 with any other protocol
USB4	-	

16.2 USB4 Router

USB4 is a Standard architecture (formerly known as CIO), but with the addition of USB 3.2 (20G) tunneling, and rounded frequencies. USB4 adds a new USB4 PD entry mode, but fully documents mode entry, and negotiation elements of Thunderbolt™ 3.

USB4 architecture (formerly known as Thunderbolt™ 3 protocol) is a transformational high-speed, dual protocol I/O, and it provides flexibility and simplicity by encapsulating both data (PCIe* & USB 3.2) and video (DisplayPort*) on a single cable connection that can daisy-chain up to five devices. USB4/Thunderbolt™ controllers act as a point of entry or a point of exit in the USB4 domain. The USB4 domain is built as a daisy chain of USB4/Thunderbolt™ enabled products for the encapsulated protocols - PCIe, USB 3.2 and DisplayPort. These protocols are encapsulated into the USB4 fabric and can be tunneled across the domain.

USB4 controllers can be implemented in various systems such as PCs, laptops and tablets, or devices such as storage, docks, displays, home entertainment, cameras, computer peripherals, high end video editing systems, and any other PCIe based device that can be used to extend system capabilities outside of the system's box.

The integrated connection maximum data rate is 20.625 Gbps per lane but supports also 20.0 Gbps, 10.3125 Gbps, and 10.0 Gbps and is compatible with older Thunderbolt™ device speeds.

16.2.1 USB4 Host Router Implementation Capabilities

The integrated USB Type-C sub-system implements the following interfaces via USB4:

- Up to two DisplayPort* sink interfaces each one capable of:
  - DisplayPort 2.1 specification for tunneling
  - 1.62 Gbps or 2.7 Gbps or 5.4 Gbps or 8.1 Gbps link rates
  - x1, x2 or x4 lane operation
  - Support for DSC compression
- Up to two PCI Express* Root Port interfaces each one capable of:
  - PCI Express* 3.0 x4 compliant @ 8.0 GT/s
- Up to two xHCI Port interfaces each one capable of:
  - USB 3.2 Gen 2x1 (10 Gbps)
  - USB 3.2 Gen 2x2 (20 Gbps)
- USB4 Host Interface:
  - PCI Express* 3.0 x4 compliant endpoint
  - Supports simultaneous transmit and receive on 12 paths
  - Raw mode and frame mode operation configurable on a per-path basis
  - MSI and MSI-X support
  - Interrupt moderation support
- USB4 Time Management Unit (TMU):
- Up to two Interfaces to USB Type-C* connectors, each one supports:
  - USB4 PD entry mode, as well as TBT 3 compatibility mode, each supporting:
    - 20 paths per port
    - Each port support 20.625/20.0 Gbps or 10.3125/10.0 Gbps link rates per lane.
    - 16 counters per port

16.3 xHCI Controllers

The processor supports xHCI controllers. The native USB 3.2 path proceeds from the memory directly to PHY.

16.3.1 USB 3 Controllers

16.3.1.1 Extensible Host Controller Interface (xHCI)

Extensible Host Controller Interface (xHCI) is an interface specification that defines Host Controller for a universal Serial Bus (USB 3.2), which is capable of interfacing with USB 1.x, 2.0, and 3.x compatible devices.

In case that a device (example, USB 3.2 Flash Drive) was connected to the computer, the computer will work as Host and the xHCI will be activated inside the processor.

The xHCI controller support link rate of up to USB 3.2 Gen 2x2 (20G).

16.3.2 PCIe Interface

Table 61. PCIe via USB4 Configuration

USB4 IPs	USB4_PCIe	USB Type-C* Ports
USB4_DMA0	USB4_PCIE0	TCP0
	USB4_PCIE1	TCP1

16.4 Display Interface

Refer to Display on page 161.

16.5 USB Type-C Signals

Signal Name	Description	Dir.	Link Type	Availability
TCP[1:0]_TX[1:0]_P TCP[1:0]_TX[1:0]_N	TX Data Lane.	O	Diff	All Processor Series
TCP[1:0]_TXRX[1:0]_P TCP[1:0]_TXRX[1:0]_N	RX Data Lane, also serves as the secondary TX data lane.	I/O	Diff	All Processor Series
TCP[1:0]_AUX_P TCP[1:0]_AUX_N	Common Lane AUX-PAD.	I/O	Diff	All Processor Series
TCP_RCOMP	Type-C Resistance Compensation.	A		All Processor Series

16.6 AUS BIAS/Orientation/Isolation Control

In USB/DP less retimer config, the AUX Bias/Orientation/Isolation Control is supported by 3rd party PD Controllers. The platform relies on the SBU XBAR mux capability inside the PD controllers. The PD controllers are responsible for switching the mux based on orientation.

17.0 Universal Serial Bus (USB)

The processor implements a standalone xHCI USB 3.2 controller which provides support for up to 8 USB 2.0 signal pairs and 2 USB 3.2 signal pairs. The xHCI controller supports wake up from sleep states S1-S4. The xHCI controller supports up to 64 devices for a maximum number of 2048 Asynchronous endpoints (Control / Bulk) or maximum number of 128 Periodic endpoints (Interrupt / isochronous).

NOTES

1. Each walk-up USB 3.2 capable port must include USB 3.2 and USB 2.0 signaling.
2. U1 and U2 Link Power Management capabilities are disabled on the xHCI controller's USB 3.2 capable ports.
3. When the processor is in package C-State C10, the standalone xHCI controller can support up to 2 concurrent traffic streams from USB 2.0 Isochronous IN Endpoints connected directly to the root port.

Table 62. Acronyms

Acronyms	Description
xHCI	eXtensible Host Controller Interface

Table 63. References

Specification	Location
USB 4.0 Specification	www.usb.org
USB 3.2 Specification	
USB 2.0 Specification	

17.1 Functional Description

17.1.1 eXtensible Host Controller Interface (xHCI) Controller

The eXtensible Host Controller Interface (xHCI) allows data transfer speed up to 10 Gbps for USB 3.2 Gen 2x1 ports , and 5 Gbps for USB 3.2 Gen 1x1 ports. The xHCI supports SuperSpeed USB 10 Gbps, SuperSpeed USB 5 Gbps, High-Speed (HS), Full-Speed (FS), and Low-Speed (LS) traffic on the bus. The xHCI supports USB Debug port on all the USB ports.

17.1.2 USB Dual Role Support - eXtensible Device Controller Interface (xDCI) Contro

The USB subsystem also supports Dual Role Capability. The xHCI is paired with a stand-alone eXtensible Device Controller Interface (xDCI) to provide dual role functionality. The USB subsystem incorporates a xDCI USB 3.2 Gen 1x1 (5 Gbps) device controller. The dual role capability supports SuperSpeed USB 5 Gbps, and High-Speed (HS) on the standalone xDCI controller. The device controller is instantiated as a separate PCI function. The USB implementation is compliant to the Device specification and supports host/device only through the integrated USB Type-C* connector.

The xDCI shares all USB ports with the host controller, with the ownership of the port being decided based the USB Power Delivery specification. Since all the ports support device mode, xDCI enabling must be extended by System BIOS and EC. While the port is mapped to the device controller, the host controller Rx detection must always indicate a disconnected port. Only one port can be connected (and active) to the device controller at one time. Any subsequent connection will not be established.

17.2 Signal Description

Signal Name	Type	Description	Availability
USB32_1_RX_N **USB32_1_RX_P**	I	**USB 3.2 Differential Receive Pair 1**: These are USB 3.2-based high-speed differential signals for Port 1. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB32_1_TX_N **USB32_1_TX_P**	O	**USB 3.2 Differential Transmit Pair 1**: These are USB 3.2-based high-speed differential signals for Port 1. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB32_2_RX_N **USB32_2_RX_P**	I	**USB 3.2 Differential Receive Pair 2**: These are USB 3.2-based high-speed differential signals for Port 2. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB32_2_TX_N **USB32_2_TX_P**	O	**USB 3.2 Differential Transmit Pair 2**: These are USB 3.2-based high-speed differential signals for Port 2. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_1 **USB2N_1**	I/O	**USB 2.0 Port 1 Transmit/Receive Differential Pair 1:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_2 **USB2N_2**	I/O	**USB 2.0 Port 2 Transmit/Receive Differential Pair 2:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All

continued...

Signal Name	Type	Description	Availability
USB2P_3 **USB2N_3**	I/O	**USB 2.0 Port 3Transmit/Receive Differential Pair 3:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_4 **USB2N_4**	I/O	**USB 2.0 Port 4 Transmit/Receive Differential Pair 4:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_5 **USB2N_5**	I/O	**USB 2.0 Port 5 Transmit/Receive Differential Pair 5:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_6 **USB2N_6**	I/O	**USB 2.0 Port 6 Transmit/Receive Differential Pair 6:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_7 **USB2N_7**	I/O	**USB 2.0 Port 7 Transmit/Receive Differential Pair 7:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
USB2P_8 **USB2N_8**	I/O	**USB 2.0 Port 8 Transmit/Receive Differential Pair 8:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	All
GPP_E09/**USB_OC0#**	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred. When configured as OC# pin, a 10 kohm pull-up resistor is required to be connected to the power-rail. When this pin is configured as GPIO, no pull-up resistor is required. *Notes:* 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	All
GPP_B14/**USB_OC1#**/ DDSP_HPDB#/DISP_MISCB/	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred. When configured as OC# pin, a 100 kohm pullup resistor is required to be connected to the power-rail. The USB_OC1# pin is multiplexed on the GPP_B14, which is a strap for Top Swap Override. A 100 kohm pull-up ensures the strap functionality is not inadvertently asserted to enable the Top Swap mode Override. When this pin is configured as GPIO, no pull-up resistor is required.	All

continued...

Signal Name	Type	Description	Availability
		Notes: 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	
USB2_1_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	All
USB2_2_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	All
USB32_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	All

17.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
USB2P_[8:1]	Internal Pull-down	14.25–24.8 kohm	1
USB2N_[8:1]	Internal Pull-down	14.25–24.8 kohm	1
Note: 1. Series resistors (45 ohm ±10%)			

17.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]
USB32_[2:1]_RX_N **USB32_[2:1]_RX_P**	Primary	Low	Low
USB32_[2:1]_TX_N **USB32_[2:1]_TX_P**	Primary	Low	Low
USB2N_[8:1]	Primary	Low	Low
USB2P_[8:1]	Primary	Low	Low
USB_OC0#	Primary	Undriven	Undriven
USB_OC1#	Primary	Undriven	Undriven
USB2_[2:1]_RCOMP	Primary	Low	Low
USB32_RCOMP	Primary	Low	Low
Note: 1. Reset reference for primary well pins is RSMRST#.			

18.0 PCI Express (PCIe)

Table 64. Acronym

Acronyms	Description
PCIe*	PCI Express* (Peripheral Component Interconnect Express*)

Table 65. Reference Table

Specification	Location
PCI Express M.2 Specification Revision 4.0, Version 1.1, April 14, 2022	https://pcisig.com/

18.1 Functional Description

Table 66. Features Supported

PCIe Controller Feature	PCIe Controllers	
	A	C
PCIe Max Rate	4.0	4.0
L1 Sub-States (L1.0, L1.1, L1.2)	Yes	Yes
L0s Link State (RX/TX)	Yes	Yes
S4/S5 Sleep States (Sx)	Yes	Yes
Common Clock Mode	Yes	Yes
Separate Reference Clock with Independent SSC (SRIS)	No	No
Separate Reference Clock with No SSC (SRNS)	No	No
Precision Time Management (PTM)	Yes	Yes
Advanced Error Reporting (AER)	Yes	Yes
End-to-End Lane Reversal	Yes	Yes
Latency Tolerance Reporting (LTR)	Yes	Yes
LTR Programmable by OS	No	No
PCIe TX Half Swing	No	No
PCIe TX Full Swing	Yes	Yes
Run Time D3 (RTD3)	Yes	Yes
RTD3 through PFET_EN to remove power from PCIe Device	Yes	Yes
Access Control Services (ACS)	Yes	Yes
Alternative Routing-ID Interpretation (ARI)	Yes	Yes
Port 8xh I/O Cycle Decode Forwarding	Yes	Yes

continued...

PCIe Controller Feature	PCIe Controllers	
	A	C
Lane Polarity Inversion	Yes	Yes
PCIe Controller Root Port Hot-Plug via CLKREQ#	Yes	Yes
Downstream Port Containment (DPC)	No	No
Enhanced Downstream Port Containment (eDPC)	No	No
Virtual Channel (VC)	0/1	0/1
NVMe Cycle Router	No	No
Volume Management Device (Intel® VMD)	No	No
RAID[0] and RAID[1] Mode Support	No	No
RAID[5] and RAID[10] Mode Support	No	No
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) Mem Write Transactions	RPs between PCA and PCC= No RPs within PCA and within PCC = Yes	
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) Mem Read Transactions	No	
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) MCTP VDM Transactions	RPs within PCA or within PCC= Yes RPs between PCA/C = Yes	
Flattening Portal Bridge (FPB)	No	No
PCIe Root Port Initiated Dynamic Width Change	No	No
PCIe Root Port Initiated Dynamic Speed Change	Yes	Yes
End Point Device Initiated Dynamic Width Change	Yes	Yes
End Point Device Initiated Dynamic Speed Change	Yes	Yes
Max Payload Size (MPS)	256B	256B

18.2 Signal Description

Signal Name	Type	Description	Processor
PCIE_A[4:1]_TX_N **PCIE_A[4:1]_TX_P**	O	**PCI Express* Controller A Differential Transmit Pairs** These are the PCI Express* based outbound high-speed differential signals from PCIe Controller A	
PCIE_C[2:1]_TX_N **PCIE_C[2:1]_TX_P**	O	**PCI Express* Controller B Differential Transmit Pairs** These are the PCI Express* based outbound high-speed differential signals from PCIe Controller C	
PCIE_A[4:1]_RX_N **PCIE_A[4:1]_RX_P**	I	**PCI Express* Controller A Differential Receive Pairs** These are the PCI Express* based inbound high-speed differential signals for PCIe Controller A	
PCIE_C[2:1]_RX_N **PCIE_C[2:1]_RX_P**	I	**PCI Express* Controller B Differential Receive Pairs** These are the PCI Express* based inbound high-speed differential signals for PCIe Controller C	
PCIE_A_RCOMP **PCIE_C_RCOMP**	A	**PCI Express* Controllers A/C PHY Impedance Compensation Inputs**	
PCIE_LINK_DOWN	O	**PCI Express* Link Down Debug Signal**	

continued...

Signal Name	Type	Description	Processor
		PCIe link failure debug signal. PCIe Root Port(s) will assert this signal when a link down event occurs and is detected. For example when a link fails to train during an L1 sub-state exit event.	

18.3 IO Signal Planes and States

Table 67. Power Plane and States for PCI Express* Signals

Signal Name	Type	Power Plane	During Reset[2]	Immediately After Reset[2]	S4/S5
PCIE_*_TX_P **PCIE_*_TX_N**	O	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
PCIE_*_RX_P **PCIE_*_RX_N**	I	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
PCIE*_RCOMP	A	Primary	Undriven	Undriven	Undriven

Notes: 1. PCIE_RXP/RXN pins transition from un-driven to Internal Pull-down during Reset.
2. Reset reference for primary well pins is RSMRST#.

18.4 PCI Express* Root Port Support Feature Details

Table 68. PCI Express* Root Port Feature Details

Processor	Max Transfer Rate	Max Devices (Root Ports)	Max Lanes	PCIe Gen Type	Encoding	Transfer Rate (MT/s)	Theoretical Max Bandwidth (GB/s)		
							x1	x2	x4
Intel® Core™ Processor (Series 3)	16 GT/s (4.0)	6	6	1	8b/10b	2500	0.25	0.50	1.00
				2	8b/10b	5000	0.50	1.00	2.00
				3	128b/130b	8000	1.00	2.00	3.94
				4	128b/130b	16000	1.97	3.94	7.88

Notes: 1. Theoretical Maximum Bandwidth (GB/s) = ((Transfer Rate * Encoding * # PCIe Lanes) /8)/1000
- PCIe 4.0 with 4 PCIe Lanes Example = ((16000 * 128/130* 4)/8)/1000 = 7.88 GB/s

2. When GbE is enabled on a PCIe* Root Port, the Max. Device (Root Ports) value listed is reduced by a factor of 1

Figure 11. Intel® Core™ Processor (Series 3) Supported PCI Express* Link Configurations

Intel® Core™ Processor (Series 3)	Platform Controller Die (PCD)					
Flex I/O Lane	**3**	**4**	**5**	**6**	**7**	**8**
PCIe Controllers	**A**				**C**	
PCIe Max Rate	**4.0**				**4.0**	
x2 Effective Throughput[4]	**2**		**2**		**1**	
PCIe (TX/RX) Lanes	**A1**	**A2**	**A3**	**A4**	**C1**	**C2**
PCIe Lane Count	**1**	**2**	**3**	**4**	**9**	**10**
PCIe Configurations (Bi-Furcation)[3]	1px4				1px2	
	1px4(LR)				1px2(LR)	
	2px2				2px1	
	2px2(LR)					
	1px2+2px1					
	1px2+2px1(LR)					
	4px1					
Logical Link Lanes	0	1	2	3	0	1
	3	2	1	0	1	0
	0	1	0	1	0	0
	1	0	1	0		
	0	1	0	0		
	0	0	1	0		
	0	0	0	0		
Assigned Root Ports	RP1				RP9	
	RP1				RP9	
	RP1		RP3		RP9	RP10
	RP3		RP1			
	RP1		RP3	RP4		
	RP4	RP3	RP1			
	RP1	RP2	RP3	RP4		
Bus - Dev - Func (BDF) Assignments[1]	**RP**	**Bus**	**Dev**	**Func**	**RP**	**Bus**	**Dev**	**Func**
	1	0h	1Ch	0h	9	0h	6h	0h
	2	0h	1Ch	1h				
	3	0h	1Ch	2h	10	0h	6h	1h
	4	0h	1Ch	3h				

NOTES

1. Device (BDF) groupings have multiple functions, the lowest active Root Port within the Device (BDF) grouping will always be assigned Function 0 while any remaining active Root Port within the Device (BDF) grouping will be assigned their mapped Function # as shown.
2. 1px2+2px1 with Lane Reversal Enabled results in a 2px1+1px2 PCIe Controller configuration.
3. Reduced Root Port width configurations, within Bi-Furcation configurations, are supported (example: x2 PCIe End Point Device populated in a PCIe Controller set as 1px4 will result in a 1px2 PCIe Root Port configuration or x1 PCIe End Point Device populated in a PCIe Controller set as 1px4 will result in a 1px1 PCIe Root Port configuration).
4. The PCIe* Link Configuration support may vary depending on the SKU. Refer to the SKU details covered in the Introduction on page 16.
5. LR = Lane Reversal
6. PCIe Configuration (#p) x (#) = (Number of PCIe Root Ports) x (Number of Data Lane Pairs per PCIe Root Port)
7. RP# refers to a specific PCI Express* Root Port #; for example RP3 = PCI Express* Root Port 3
8. A PCIe* Lane is composed of a single pair of Transmit (TX) and Receive (RX) differential pairs. A connection between two PCIe* devices is known as a PCIe* Link, and is built up from a collection of one or more PCIe* Lanes which make up the width of the link (such as bundling 2 PCIe* Lanes together would make a x2 PCIe* Link). A PCIe* Link is addressed by the lowest number PCIe* Lane it connects to and is known as the PCIe* Root Port (such as a x2 PCIe* Link connected to PCIe* Lanes 3 and 4 would be called x2 PCIe* Root Port 3).
9. The PCIe* Lanes can be configured independently from one another but the max number of configured Root Ports (Devices) must not be exceeded
10. Unidentified lanes within a PCIe* Link Configuration are disabled but their physical lanes are used for the identified Root Port

19.0 Universal Flash Storage

Universal Flash Storage is only supported on NVL 6C, 8C, and H16C SKUs.

The Universal Flash System (UFS) is the next generation storage standard. UFS defines a unique feature set that provides low power consumption, high data throughput, low electromagnetic interference and optimization for mass memory subsystem efficiency.

- UFS controller is UFS 4.0 (JESD220F) compliant.
- UFS supports the OS to boot and can be used as either primary or secondary storage.
- UFS host controller supports UFS inline encryption.

NOTE

The initial boot process uses SPI Flash for storing and loading IFWI.

19.1 UFS Functional Description

UFS adopts the latest technology from below industrial standards,

- MIPI M-PHY specification v5.0
- MIPI UniPro specification v2.0
- INCTS T10 SCSI standards (SBC, SPC and SAM) for command sets and architecture model

NVL conforms to JEDEC UFS Host Controller Interface Specification JESD220F. The UFS host controller provides an interface method of accessing the UFS hardware capabilities. The UFS Host Controller handles UFS Protocol at transmission level, packing data, adding cyclic redundancy check (CRC), start/end bit, and checking for transaction format correctness.

19.2 UFS Signals

Signal Name	Type	Description
UFS0_TX_P	O	UFS port lane 0 transmit signal
UFS0_TX_N	O	UFS port lane 0 transmit signal
UFS0_RX_P	I	UFS port lane 0 receive signal
UFS0_RX_N	I	UFS port lane 0 receive signal
UFS1_TX_P	O	UFS port lane 1 transmit signal
UFS1_TX_N	O	UFS port lane 1 transmit signal

continued...

Signal Name	Type	Description
UFS1_RX_P	I	UFS port lane 1 receive signal
UFS1_RX_N	I	UFS port lane 1 receive signal
UFS_RCOMP	Analog	UFS Resistor Compensation
GPP_D21/**UFS_REFCLK**/ SRCCLKREQ8#	IOD	UFS Reference Clock

19.3 IO Signals Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S4/S5
UFS[0:1]_TX_P UFS[0:1]_TX_N UFS[0:1]_RX_P UFS[0:1]_RX_N	Primary	Undriven	Undriven	Undriven
UFS_REFCLK	Primary	Undriven	Undriven	Undriven

20.0 Graphics

20.1 Processor Graphics

The processor graphics is based on Xe3 graphics core architecture that enables substantial gains in performance and lower-power consumption over prior generations. Xe3 architecture supports up to 2 Xe3-core depending on the processor SKU.

The processor graphics architecture delivers high dynamic range of scaling to address segments spanning low power to high power, increased performance per watt, support for next generation of APIs. Xe3 scalable architecture is partitioned by usage domains along Render/Geometry, Media, and Display. The architecture also delivers very low-power video playback and next generation analytics and filters for imaging related applications. The new Graphics Architecture includes 3D compute elements, Multi-format HW assisted decode/encode pipeline, and Mid-Level Cache (MLC) for superior high definition playback, video quality, and improved 3D performance and media.

20.1.1 Media Support (Intel® QuickSync and Clear Video Technology HD)

Xe3 implements multiple media video codecs in hardware as well as a rich set of image processing algorithms.

20.1.1.1 Hardware Accelerated Video Decode

Xe3 implements a high-performance and low-power HW acceleration for video decoding operations for multiple video codecs.

The HW decode is exposed by the graphics driver using the following APIs:

- Direct3D11 Video API
- Direct3D12 Video API
- Intel Media SDK
- MFT (Media Foundation Transform) filters[1]
- Intel VA API [2]
- Intel one VPL

NOTES

1. Only for JPEG Decoder
2. Only for Linux*

Xe3 supports full HW accelerated video decoding for MPEG2/AVC/HEVC/VP9/JPEG/AV1.

Table 69. Hardware Accelerated Video Decoding

Codec	Profile	Level	Maximum Resolution
MPEG2	Main	Main - 15Mbps High - 40Mbps	FHD
AVC/H264	High Main Constrained Baseline	L5.2	4K
	4:2:0 8bit 4:2:0/4:2:2 10bit		4K @ 60
JPEG/MJPEG	Baseline	Unified level	16K x16K
HEVC/H265	**Main12** 420, 422, 444 - 8b/10b/12b SCC 420, 444 - 8b/10b	L6.1	4k @60(Decode Only) 4k @60(Decode Playback)
VP9	8b 420/444 (NV12/AYUV) 10b 420/444 (P010/Y410) 12b 420/444 (P016/Y416)	Unified level	4k @60(Decode only) 4k @60(Decode Playback)
VP8	8b 420	Unified level	4k @60(Decode only) 4k @60(Decode Playback)
AV1	Main10 (420, 444 8b/10b)	L6.1	4k @60 (Video, Decode only) 4k @60 (Decode Playback)

Expected Performance: More than TBD simultaneous decode streams @ 1080p.

VVC codec: Not supported

NOTE

Actual performance depends on the processor SKU, content bit rate, and memory frequency. Hardware decode for H264 SVC is not supported. Hardware Decide for VVC is not supported.

20.1.1.2 Hardware Accelerated Video Encode

Xe3 implements a low-power low-latency fixed function encoder which supports AVC, HEVC, VP9, JPEG, and AV1.

The HW encode is exposed by the graphics driver using the following APIs:

- Direct3D12 Video API
- Intel®one VPL
- MFT (Media Foundation Transform) filters [Only for AVC/HEVC/JPEG/AV1 Encoder]

Xe3 supports full HW accelerated video encoding for AVC/HEVC/VP9/JPEG/AV1.

Table 70. Hardware Accelerated Video Encode

Codec	Profile	Level	Maximum Resolution
AVC/H264	High Main	L5.2	4K@60

continued...

Codec	Profile	Level	Maximum Resolution
	Constrained Baseline		
JPEG			16Kx16K
HEVC/H265	Main10 422 , 444 - 8b/10b	L5.2	4K@60
AV1 encode	**Main** : 420 8b/10b **High** : 444 8b/10b	**L5.1**	4k@60
VP9 encode	**Profile 0** : 420 8b **Profile 1** : 444 8b **Profile 2** : 420 10b **Profile 3** : 444 10b	L5.1	4k@60

NOTE

Hardware encode for H264 SVC is not supported.

VVC codec: Not supported

20.1.1.3 Hardware Accelerated Video Processing

20.1.1.4 Hardware Accelerated Transcoding

Transcoding is a combination of decode, video processing (optional) and encode. Using the above hardware capabilities can accomplish a high-performance transcode pipeline. There is not a dedicated API for transcoding.

The processor graphics supports the following transcoding features:

- High performance high quality flexible encoder for video editing, video archiving.
- Low-power low latency encoder for video conferencing, wireless display, and game streaming.
- Low power Scaler and Format Converter.

20.1.2 Graphics Core Cache

The Xe3 Graphics Core architecture has a hierarchy of caches which contains first, second and third level caches.

First and Second Level Cache

The first and second level cache is a lower-level Instruction and the Data caches. They are implemented close to the Xe3/3D compute elements, decode/encode and media pipelines. These cache units are not shared between the different units.

Third Level Cache

Third level cache is a central memory cache that is implemented as higher cache hierarchy. The device cache is a multi-way set-associative that allow memory pages to be cached either coherently or non-coherently with respect to an external memory system.

20.2 Platform Graphics Hardware Feature

20.2.1 Hybrid Graphics

Microsoft* Windows* 11 operating system enables the Windows*11 Hybrid graphics framework wherein the GPUs and their drivers can be simultaneously utilized to provide users with the benefits of both performance capability of discrete GPU (dGPU) and low-power display capability of the processor GPU (iGPU). For instance, when there is a high-end 3D gaming workload in progress, the dGPU will process and render the game frames using its graphics performance, while iGPU continues to perform the display operations by compositing the frames rendered by dGPU. We recommend that OEMS should seek further guidance from Microsoft* to confirm that the design fits all the latest criteria defined by Microsoft* to support HG.

Microsoft* Hybrid Graphics definition includes the following:

1. The system contains a single integrated GPU and a single discrete GPU.
2. It is a design assumption that the discrete GPU has a significantly higher performance than the integrated GPU.
3. Both GPUs shall be physically enclosed as part of the system.
   a. Microsoft* Hybrid DOES NOT support hot-plugging of GPUs
   b. OEMS should seek further guidance from Microsoft* before designing systems with the concept of hot-plugging
4. Starting with Windows*11 (WDDM 2.0), a previous restriction that the discrete GPU is a render-only device, with no displays connected to it, has been removed. A render-only configuration with NO outputs is still allowed, just NOT required.

21.0 Display

21.1 Display Technologies Support

Technology	Standard
eDP* 1.5	VESA* Embedded DisplayPort* Standard 1.5
DisplayPort* 2.1	VESA Certified DisplayPort (**v2.1**, HBR3, DSC)
HDMI* 2.1	High-Definition Multimedia Interface Specification Version 2.1

Table 71. Display Ports Availability and Link Rate

Port	Intel® Core™ Series 3 Supported Technologies
DDI A	eDP upto HBR3 8.1Ghz
DDI B	HDMI TMDS 6Gbps
TCP 0	DP over Type-C up to HBR3 8.1Ghz
TCP 1	DP over Type-C up to HBR3 8.1Ghz
Note: DDIA must be used as the eDP port. TCP port do not support *native*DP, only DPoC is POR.	

21.2 Display Interfaces

21.2.1 Digital Display Interface DDI Signals

Signal Name	Type	Description
DDIA_TX_P[3:0] DDIA_TX_N[3:0]	O	Digital Display Interface A (DDIA): Digital Display Interface main link transmitter lanes.
DDIA_AUX_P DDIA_AUX_N	I/O	Digital Display Interface A (DDIA): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.
GPP_E08/**DDPA_CTRLDATA** GPP_E07/**DDPA_CTRLCLK**	I/O	Digital Display Interface A (DDIA): HDMI Graphics Management Bus (GMBUS).
DDSP_HPDA#	I	Digital Display Interface A (DDIA): Hot Plug Detect (HPD).
VDDEN	O	Digital Display Interface A (DDIA): eDP Panel power control enable signa
BKLTEN	O	Digital Display Interface A (DDIA): eDP Panel back-light control enable signal.
BKLTCTL	O	Digital Display Interface A (DDIA): eDP Panel back-light control Pulse Wide Modulation (PWM) signal.
DDIB_TX_P[3:0] DDIB_TX_N[3:0]	O	Digital Display Interface A (DDIB): Digital Display Interface main link transmitter lanes.

continued...

Signal Name	Type	Description
GPP_C23/**DDPB_CTRLDATA** GPP_C22/**DDPB_CTRLCLK**	O	Digital Display Interface A (DDIB): HDMI Graphics Management Bus (GMBUS).
GPP_B14/USB_OC1#/**DDSP_HPDB#**/DISP_MISCB	I	Digital Display Interface A (DDIB): Hot Plug Detect (HPD).
DDI_RCOMP	Analog	DDI IO Compensation resistors.
Notes: • Auxiliary Channel (AUX CH) is a half-duplex bidirectional channel used for link management and device control. AUX CH is an AC coupled differential signal. • GMBUS follows I2C Protocol.		

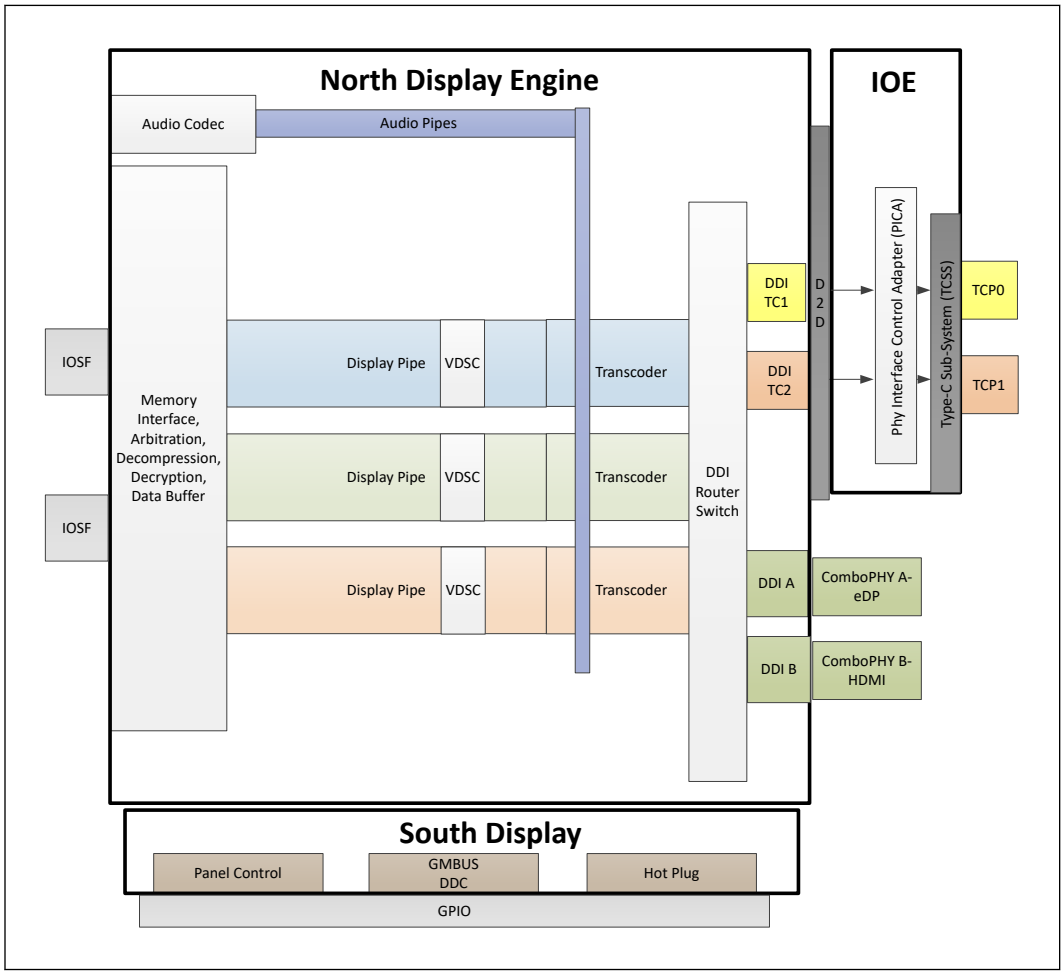
21.2.2 Digital Display Interface TCP Signals

Signal Name	Type	Description
TCP0_TXRX[1:0]_P TCP0_TXRX[1:0]_N TCP0_TX[1:0]_P TCP0_TX[1:0]_N	O	Digital Display Interface 0 (TCP0): Digital Display Interface main link transmitter lanes.
TCP0_AUX_P TCP0_AUX_N	I/O	Digital Display Interface 0 (TCP0): DisplayPort Auxiliary: Half-duplex, bidirectional channel consists of one differential pair.
GPP_C17/TBT_LSX0_RXD/**DDP0_CTRLDATA** GPP_C16/TBT_LSX0_TXD/**DDP0_CTRLCLK**	I/O	Digital Display Interface 0 (TCP0): HDMI Graphics Management Bus (GMBUS).
GPP_B09/**DDSP_HPD0#**/DISP_MISC1	I	Digital Display Interface 0 (TCP0): Hot Plug Detect (HPD).
TCP1_TXRX[1:0]_P TCP1_TXRX[1:0]_N TCP1_TX[1:0]_P TCP1_TX[1:0]_N	O	Digital Display Interface 1 (TCP1): Digital Display Interface main link transmitter lanes.
TCP1_AUX_P TCP1_AUX_N	I/O	Digital Display Interface 1 (TCP1): DisplayPort Auxiliary: Half-duplex, bidirectional channel consists of one differential pair.
GPP_C19/TBT_LSX1_RXD/**DDP1_CTRLDATA** GPP_C18/TBT_LSX1_TXD/**DDP1_CTRLCLK**	I/O	Digital Display Interface 1 (TCP1): HDMI Graphics Management Bus (GMBUS).
GPP_B10/**DDSP_HPD1#**/DISP_MISC2	I	Digital Display Interface 1 (TCP1): Hot Plug Detect (HPD).
TCP_RCOMP	Analog	DDI IO Compensation resistors.
GPP_B09/DDSP_HPD0#/**DISP_MISC1**	O	Display Misc signals.
GPP_B10/DDSP_HPD1#/**DISP_MISC2**	O	Display Misc signals.
GPP_B14/USB_OC1#/DDSP_HPDB#/**DISP_MISCB**	O	Display Misc signals.
Notes: • Auxiliary Channel (AUX CH) is a half-duplex bidirectional channel used for link management and device control. AUX CH is an AC coupled differential signal. • GMBUS follows I2C Protocol		

21.3 Display Features

21.3.1 General Capabilities

Figure 12. Processor Display Architecture



NOTE

For port availability in each of the processor lines, refer to Table 71 on page 161.

- Up to 3 simultaneous displays, 3840x2160 60HDR Embedded panel concurrent with:
  - — Up to 2x 3840x2160 60HDR External panels.

  NOTE

  Maximum resolution capability can only be supported with 2 channel maximum memory speeds as defined in Processor SKU Support Matrix on page 131.

  Display interfaces supported:

  - — DDI interface supports eDP*
  - — DDI interface supports HDMI
  - — TCP interfaces - no support for Native HDMI/DP, support, DP* Alt mode and DP* tunneled.
- End-To-End (E2E) compression, Unified memory compression across GT, media and display.
- Audio stream support on external ports.
- HDR (High Dynamic Range) support.
- Three Display Pipes - Supporting blending, color adjustments, scaling and dithering.
- Transcoder - Containing the Timing generators supporting eDP*, DP*, HDMI* interfaces.
- Power optimized pipe supporting Embedded DisplayPort*
  - — FBC (Frame Buffer Compression) - power saving feature.

21.3.2 Multiple Display Configurations

The following multiple display configuration modes are supported (with appropriate driver software):

- Single Display is a mode with one display port activated to display the output to one display device.
- Display Clone is a mode with up to three display ports activated to drive the display content of same color depth setting but potentially different refresh rate and resolution settings to all the active display devices connected.
- Extended Desktop is a mode with up to three display ports activated to drive the content with potentially different color depth, refresh rate, and resolution settings on each of the active display devices connected.

21.3.3 High-bandwidth Digital Content Protection (HDCP)

HDCP is the technology for protecting high-definition content against unauthorized copy or unreceptive between a source (computer, digital set top boxes, and so on) and the sink (panels, monitor, and TVs). The processor supports both HDCP 2.3 and HDCP 1.4 content protection over wired displays (HDMI* and DisplayPort*).

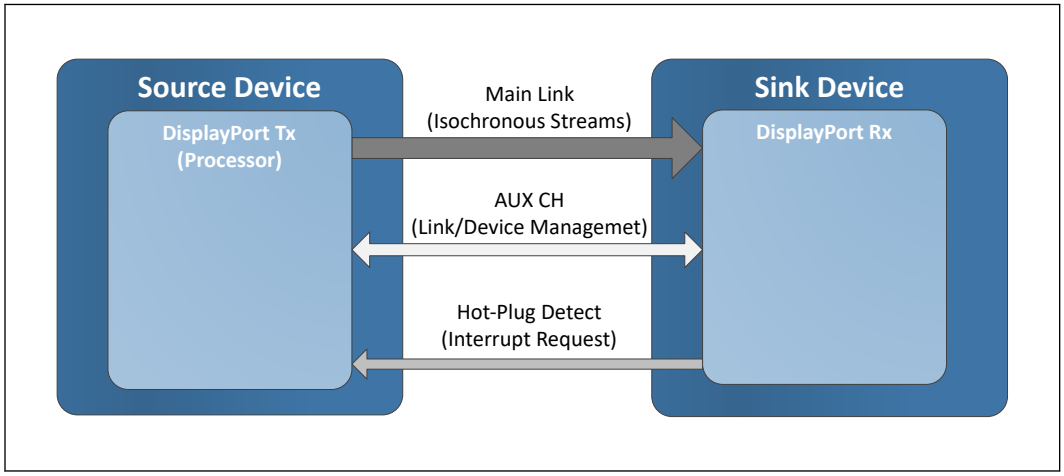
21.3.4 DisplayPort

The DisplayPort* is a digital communication interface that uses differential signaling to achieve a high-bandwidth bus interface designed to support connections between PCs and monitors, projectors, and TV displays.

A DisplayPort* consists of a Main Link (four lanes), Auxiliary channel, and a Hot-Plug Detect signal. The Main Link is a unidirectional, high-bandwidth, and low-latency channel used for transport of isochronous data streams such as uncompressed video and audio. The Auxiliary Channel (AUX CH) is a half-duplex bi-directional channel used for link management and device control. The Hot-Plug Detect (HPD) signal serves as an interrupt request from the sink device to the source device.

The processor is designed in accordance with VESA* DisplayPort* specification.

Figure 13. DisplayPort* Overview



- Support main link of 1, 2, or 4 data lanes.
- Link rate support up to HBR3 /8.1Ghz
- Aux channel for Link/Device management.
- Hot Plug Detect.
- Support up to 36 BPP (Bit Per Pixel).
- Support SSC.
- Support YCbCR 4:4:4, YCbCR 4:2:0, YCbCR 4:2:2, and RGB color format.
- Support MST (Multi-Stream Transport).
- Support VESA DSC 1.2a.
- Support panel replay.
- Adaptive Sync.

21.3.4.1 Multi-Stream Transport (MST)

- The processor supports Multi-Stream Transport (MST), enabling multiple monitors to be used via a single DisplayPort connector.
- The processor supports maximum 3 MST concurrent displays. when eDP is OFF (Lid closed) and HDMI* on DDIB is Off.
- The processor supports maximum 2 MST concurrent displays. when eDP is On. and HDMI* on DDIB is Off.
- The processor supports maximum 1 MST concurrent displays. when eDP is On. and HDMI* on DDIB is On.
- Maximum MST DP supported resolution:

Table 72. Display Resolutions and Link Bandwidth for Multi-Stream Transport Calculations

Pixels per Line	Lines	Refresh Rate [Hz]	Pixel Clock [MHz]	Link Bandwidth [Gbps]
1920	1080	60	148.5	4.46
1920	1200	60	154	4.62
2048	1152	60	156.75	4.70
2048	1280	60	174.25	5.23
2048	1536	60	209.25	6.28
2304	1440	60	218.75	6.56
2560	1440	60	241.5	7.25
3840	2160	30	262.75	7.88
2560	1600	60	268.5	8.06
2880	1800	60	337.5	10.13
3200	2400	60	497.75	14.93
3840	2160	60	533.25	16.00

Note: 1. All the above is related to bit depth of 24 and can also support 30bit color.
2. The data rate for a given video mode can be calculated as:
   Data Rate = Pixel Frequency * Bit Depth
3. The bandwidth requirements for a given video mode can be calculated as:
   Bandwidth = Data Rate * 1.25 (for 8b/10b coding overhead) for DP1.4b upto HBR3
4. The link bandwidth depends on the standards is reduced blanking or not.
   If the standard is not reduced blanking - the expected bandwidth may be higher.
   For more details, refer to VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMT). Version 1.0, Rev. 13 February 8, 2013
5. To calculate what are the resolutions that can be supported in MST configurations, follow the below guidelines:
   a. Identify what is the link bandwidth column according to the requested display resolution.
   b. Summarize the bandwidth for two of three displays accordingly, and make sure the final result is below 16.2 Gbps.
      For example:
      Docking two displays: 1920x1080@120Hz +1920x1080@120Hz =7.5 +7.5 = 15 [Supported without Display Stream compression]

Table 73. DisplayPort Maximum Resolution

Standard	Intel® Core™ Processor (Series 3) Processor Series
DP*	4K60

Notes: 1. bpp - bit per pixel.
2. Resolution support is subject to memory BW availability.

21.3.5 High-Definition Multimedia Interface (HDMI)

The High-Definition Multimedia Interface (HDMI*) is provided for transmitting digital audio and video signals from DVD players, set-top boxes, and other audio-visual sources to television sets, projectors, and other video displays. It can carry high-quality multi-channel audio data and all standard and high-definition consumer

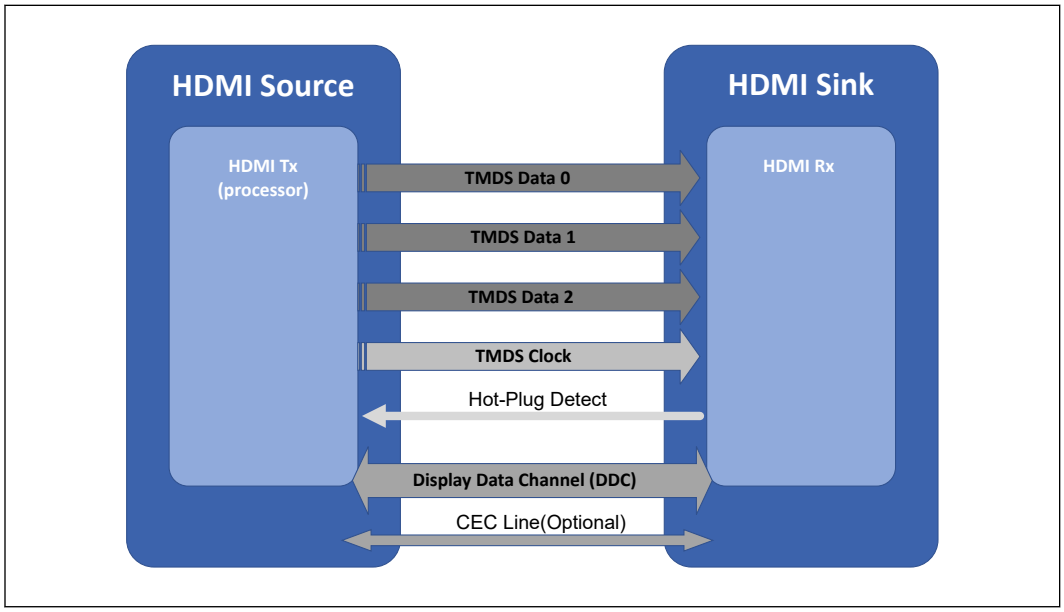
electronics video formats. The HDMI display interface connecting the processor and display devices uses transition minimized differential signaling (TMDS) to carry audiovisual information through the same HDMI cable.

HDMI* includes three separate communications channels: TMDS, DDC/GMBUS, and the optional CEC (consumer electronics control). CEC is not supported on the processor. As shown in the following figure, the HDMI* cable carries four differential pairs that make up the TMDS data and clock channels or FRL lanes. These channels are used to carry video, audio, and auxiliary data. In addition, HDMI carries a VESA DDC. The DDC/GMBUS is used by an HDMI* Source to determine the capabilities and characteristics of the Sink.

Audio, video, and auxiliary (control/status) data is transmitted across the three TMDS data channels.

In TMDS mode, The video pixel data is transmitted via TMDS clock and data lanes and is used by the receiver for data recovery on the three data channels.

Figure 14. HDMI* Overview



- Support up to 6Gbps TMDS link rates on 3 lanes.
- Support YCbCR 4:4:4, YCbCR 4:2:0, YCbCR 4:2:2, and RGB color format.
- Support up to 36 BPP (Bit Per Pixel).
- Support VESA DSC 1.2a in FRL mode.
- Hot Plug Detect.
- Variable Refresh Rate (VRR) supported.

Table 74. HDMI Maximum Resolution

Standard	Intel® Core™ Processor (Series 3) Processor Series
HDMI 2.1 TMDS (Up to 6Gbps)	4K60 Hz 24 bpp
Notes: 1. bpp - bit per pixel. 2. Resolution support is subject to memory BW availability. 3. Compressed mean DSC only	

21.3.6 embedded DisplayPort (eDP)

The embedded DisplayPort* (eDP*) is an embedded version of the DisplayPort standard oriented towards applications such as notebook and All-In-One PCs. Like DisplayPort, embedded DisplayPort* also consists of the Main Link, Auxiliary channel, and an optional Hot-Plug Detect signal.

- Support on Low power optimized pipes.
- Support up to HBR3 link rate.
- Support Backlight PWM control and enable signals, and power enable.
- Support VESA DSC 1.2a.
- Support SSC.
- Panel Self Refresh 1.
- Panel Self Refresh 2.
- MSO 2x2, 4x1(Multi Segment Operation).
- Dedicated Aux channel.
- Adaptive Sync.

eDP1.5 Supported Features:

- Early Transport
- Link-off between active frames non-PSR
- PanelReplay + ALPM
- Selective Update + Early Transport + PanelReplay + ALPM
- Aux-Less ALPM
- Panel Replay + Adaptive Sync

Table 75. Embedded DisplayPort Maximum Resolution

Standard	Intel® Core™ Processor (Series 3) Processor Series[1]
eDP*	4K60Hz HDR
Notes: 1. Maximum resolution is based on the implementation of 4 lanes at HBR3 link data rate. 2. Resolution support is subject to memory BW availability. 3. High resolution panels supporting Display Stream Compression (DSC) are supported, technology enablement may be limited due to low market availability.	

21.3.7 Integrated Audio

- HDMI* and DisplayPort interfaces can carry audio along with video.

- The processor supports up to four High Definition Audio streams on four digital ports simultaneously.

Table 76. Processor Supported Audio Formats over HDMI* and DisplayPort*

Audio Formats	HDMI*	DisplayPort*
AC-3 Dolby* Digital	Yes	Yes
Dolby* Digital Plus	Yes	Yes
DTS-HD*	Yes	Yes
LPCM, 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, and 192KHz, 16/24 bit, 2/4/6/8 channels	Yes	Yes
Dolby* TrueHD, DTS-HD Master Audio* (Lossless Blu-Ray Disc* Audio Format)	Yes	Yes

The processor will continue to support Silent stream. A Silent stream is an integrated audio feature that enables short audio streams, such as system events to be heard over the HDMI* and DisplayPort* monitors. The processor supports silent streams over the HDMI and DisplayPort interfaces at 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, and 192KHz sampling rates and silent multi-stream support.

22.0 Processor Sideband Signals

The sideband signals are used for the communication between the interfaces within the processor.

22.1 Signal Description

Signal Name	Type	Description
THERMTRIP#	O	Signal from the processor to indicate that a thermal overheating has occurred.
CATERR#	O	Signal from the processor to indicate that the system has experienced a catastrophic error and cannot continue to operate.
FORCEPR#	I	Signal from the processor to indicate the processor has reached its maximum safe operating temperature.
VIDSOUT	I/O	Signal used to transfer power management information between the processor and the voltage regulator controllers.
VIDSCK	O	Signal used to transfer power management information between the processor and the voltage regulator controllers.
VIDALERT#	I	Signal used to transfer power management information between the processor and the voltage regulator controllers.
GPP_E03/**PROC_GP0**	I	Thermal management signal
GPP_D03/**PROC_GP1**	I	Thermal management signal
GPP_E01/**PROC_GP2**/RSVD/ISH_GP5A	O	Thermal management signal
GPP_E02/**PROC_GP3**/VRALERT#/ ISH_GP10	I	Thermal management signal

NOTE

If THERMTRIP# goes active, the processor is indicating an overheat condition. PROC_GP can be used from external sensors for the thermal management.

22.2 Integrated Pull-Ups and Pull-Downs

None

22.3 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset [1]	S4/S5
THERMTRIP#	Primary	Undriven	Undriven	OFF
PROC_GP[3:0]	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

23.0 General Purpose Input and Output

The General Purpose Input/Output (GPIO) signals are grouped into multiple groups (such as GPP_A, GPP_B, and so on). All GPIO groups are powered by the Primary well.

The high level features of GPIO:

- 1.8 V operation (including the muxed functions on the pin).
- Integrated pull-up / pull-down.
- Configurable as GPIO input, GPIO output, or native function signal.
- Configurable GPIO pad ownership by host, CSME, or ISH.
- SCI (GPE) and IOAPIC interrupt capable on all GPIOs
- NMI and SMI capability capable (on selected GPIOs).
- PWM, Serial Blink capable (on selected GPIOs).

Table 77. Acronyms

Acronyms	Description
GPI	General Purpose Input
GPO	General Purpose Output
GPP	General Purpose I/O in Primary Well

23.1 Functional Description

23.1.1 Timed GPIO

The processor supports two Timed GPIOs as native function (TIME_SYNC) that is multiplexed on GPIO pins. The intent usage of the Timed GPIO function is for time synchronization purpose.

Timed GPIO can be an input or an output:

- As an input, a GPIO input event triggers the HW to capture the processor Always Running Timer (ART) time in the Time Capture register. The GPIO input event must be asserted for at least two crystal oscillator clocks period in order for the event to be recognized.
- As an output, a match between the ART time and the software programmed time value triggers the HW to generate a GPIO output event and capture the ART time in the Time Capture register. If periodic mode is enabled, HW generates the periodic GPIO events based on the programmed interval. The GPIO output event is asserted by HW for at least two crystal oscillator clock periods.

NOTE

TIME_SYNC can be set as input when both Direction (DIR) bit and Enable (EN) bit in Timed GPIO Control Register are set to 1 (refer to Datasheet Vol2 for the register info). When EN bit is set to 0, TIME_SYNC will default to output low regardless of DIR bit setting.

Timed GPIO supports event counter. When Timed GPIO is configured as input, event counter increments by one for every input event triggered. When Timed GPIO is configured as output, event counter increments by one for every output event generated. The event counter provides the correlation to associate the Timed GPIO event (the nth event) with the captured ART time. The event counter value is captured when a read to the Time Capture Value register occurs.

NOTE

When Timed GPIO is enabled, the crystal oscillator will not be shut down as crystal clock is needed for the Timed GPIO operation. As a result, SLP_S0# will not be asserted. This has implication to platform power (such as IDLE or S0ix power). Software should only enable Timed GPIO when needed and disable it when Timed GPIO functionality is not required.

23.2 Signal Description

For GPIO pin implementation including multiplexed native functions, default values, signal states, and other characteristics, download the pdf, click on the navigation pane and refer the spreadsheet, 913965-001_GPIO.xlsx..

24.0 Interrupt Timer Subsystem (ITSS)

Table 78. Acronyms

Acronym	Description
ITSS	Interrupt Timer Subsystem
HPET	High Precision Event Timer
8254 PIT	Legacy 8254 Programmable Interrupt Timer
INTR	Interrupt
NMI	Non-maskable Interrupt
INIT	Processor Initialization
SERR	System Error

Table 79. References

Specification	Document Number/Location
ACPI Specification, Rev 5.0a	https://uefi.org/acpi/specs

24.1 Feature Overview

ITSS supports following features:

- It houses the HPET, Legacy 8254 Timers and APIC Interrupt Controllers.
- Fully synchronous-based design adopted for 8254 PIT.
- Functions as a simple Internal Host Space Error Collector and Reporting Block.
- 8254 PIT - Consists of 3 16-bit Timers capable of supporting up to 6 different modes.
- APIC - Supports up to 120 IRQs.
- HPET - Contains 8 Timer Blocks and a single always running 64-bit counter. Each Timer is interrupt capable, with option to route to APIC or directly to hose using MSI. Improved resolution, reduced overhead in comparison to Legacy 8254, IOxAPIC & RTC Timers.

24.2 Functional Description

The ITSS (Interrupt Timer Sub System) have below sub blocks:

- **ITSS** : Consists of the HPET, 8254 and APIC.

24.2.1 8254 Timers

There are three counters that have fixed uses. All registers and functions associated with these timers are in the Primary well. The 8254 unit is clocked by a 1.193 MHz periodic timer tick, which is functional only in S0 states. The 1.193 MHz periodic timer tick is generated off the XTAL clock.

Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is typically programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value 1 counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two each counter period. The counter negates IRQ0 when the count value reaches 0. It then reloads the initial count value and again decrements the initial count value by two each counter period. The counter then asserts IRQ0 when the count value reaches 0, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

24.2.1.1 Timer Programming

The counter/timers are programmed in the following fashion:

1. Write a control word to select a counter.
2. Write an initial count for that counter.
3. Load the least and/or most significant bytes (as required by Control Word bits 5, 4) of the 16 bit counter.
4. Repeat with other counters.

Only two conventions need to be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant Byte only, most significant Byte only, or least significant Byte, and then most significant Byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read/write 2-byte counts, the following precaution applies – a program must not transfer control between writing the first and second Byte to another routine, which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of counter. Several commands are available:

- **Control Word Command:** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command:** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command:** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

The table below lists the six operating modes for the interval counters:

Table 80. Counter Operating Modes

Mode	Function	Description
0	Out signal on end of count (=0)	Output is 0. When count goes to 0, output goes to 1 and stays at 1 until counter is reprogrammed.
1	Hardware retriggerable one-shot	Output is 0. When count goes to 0, output goes to 1 for one clock time.
2	Rate generator (divide by n counter)	Output is 1. Output goes to 0 for one clock time, then back to 1 and counter is reloaded.
3	Square wave output	Output is 1. Output goes to 0 when counter rolls over, and counter is reloaded. Output goes to 1 when counter rolls over, and counter is reloaded, and so on
4	Software triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.
5	Hardware triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.

24.2.1.2 Reading from Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters—a simple read operation, counter Latch command, and the Read-Back command. Each one is explained below:

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for 2-byte counts, 2-bytes must be read. The2-bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

Simple Read

The first method is to perform a simple read operation. The counter is selected through Port 40h (Counter 0).

NOTE

Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations.

Counter Latch Command

The Counter Latch command, written to Port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a 2-byte count. The count value is then read from each counter's Count register as was programmed by the Control register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

If a Counter is latched and then, sometime later, latched again before the count is read, the second Counter Latch command is ignored. The count read is the count at the time the first Counter Latch command was issued.

Read Back Command

The Read Back command, written to Port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both the count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. If multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both the count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, returns the latched count. Subsequent reads return unlatched count.

24.2.2 APIC Advanced Interrupt Controller

The APIC is accessed via an indirect addressing scheme. These registers are mapped into memory space. These are programmable through PCI Config IOAC register. Refer to the Datasheet Vol.2 for more details.

24.2.3 High Precision Event Timer (HPET)

This function provides a set of timers that can be used by the operating system. The timers are defined such that the operating system may assign specific timers to be used directly by specific applications. Each timer can be configured to cause a separate interrupt.

The processor provides eight timers. The timers are implemented as a single counter with a set of comparators. Each timer has its own comparator and value register. The counter increases monotonically. Each individual timer can generate an interrupt when the value in its value register matches the value in the main counter.

Timer 0 supports periodic interrupts.

The registers associated with these timers are mapped to a range in memory space (much like the I/O APIC). However, it is not implemented as a standard PCI function. The BIOS reports to the operating system the location of the register space using

ACPI. The hardware can support an assignable decode space; however,BIOS sets this space prior to handing it over to the operating system. It is not expected that the operating system will move the location of these timers once it is set by BIOS.

Table 81. References

Specification	Location
IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a	https://www.intel.com/content/dam/www/public/us/en/documents/technical-specifications/software-developers-hpet-spec-1-0a.pdf

24.2.3.1 Timer Accuracy

The timers are accurate over any 1 ms period to within 0.05% of the time specified in the timer resolution fields.

Within any 100 us period, the timer reports a time that is up to two ticks too early or too late. Each tick is less than or equal to 100 ns; thus, this represents an error of less than 0.2%.

The timer is monotonic. It does not return the same value on two consecutive reads (unless the counter has rolled over and reached the same value).

The main counter uses the XTAL as its clock. The accuracy of the main counter is as accurate as the crystal that is used in the system. The XTAL clock frequency is determined by the pin strap that is sampled on RSMRST#.

24.2.3.2 Timer Off-load

The timer off-load feature allows the HPET timers to remain operational during very low power S0 operational modes when the XTAL clock is disabled. The clock source during this off-load is the Real Time Clock's 32.768 kHz clock. This clock is calibrated against the XTAL clock during boot time to an accuracy that ensures the error introduced by this off-load is less than 10 ppb (0.000001%).

When the XTAL clock is active, the 64 bit counter will increment by one each cycle of the XTAL clock when enabled. When the XTAL clock is disabled, the timer is maintained using the RTC clock. The long-term (> 1 ms) frequency drift allowed by the HPET specification is 500 ppm. The off-load mechanism ensures that it contributes < 1 ppm to this, which will allow this specification to be easily met given the clock crystal accuracies required for other reasons.

Timer off-load is prevented when there are HPET comparators active.

The HPET timer runs typically on the XTAL crystal clock and is off-loaded to the 32 kHz clock once the processor enters C10. This is the state where there are no C10 wake events pending and when the off-load calibrator is not running. HPET timer re-uses this 28 bit calibration value calculated by PMC when counting on the 32 kHz clock. During C10 entry, PMC sends an indication to HPET to off-load and keeps the indication active as long as the processor is in C10 on the 32 kHz clock. The HPET counter will be off-loaded to the 32 kHz clock domain to allow the XTAL clock to shut down when it has no active comparators.

Theory of Operation

The Off-loadable Timer Block consists of a 64 bit fast clock counter and an 82 bit slow clock counter. During fast clock mode the counter increments by one on every rising edge of the fast clock. During slow clock mode, the 82 bit slow clock counter will increment by the value provided by the Off-load Calibrator.

The Off-loadable Timer will accept an input to tell it when to switch to the slow RTC clock mode and provide an indication of when it is using the slow clock mode. The switch will only take place on the slow clock rising edge, so for the 32 kHz RTC clock the maximum delay is around 30 us to switch to or from slow clock mode. Both of these flags will be in the fast clock domain.

When transitioning from fast clock to slow clock, the fast clock value will be loaded into the upper 64 bits of the 82 bit counter, with the 18 LSBs set to zero. The actual transition though happens in two stages to avoid metastability. There is a fast clock sampling of the slow clock through a double flop synchronizer. Following a request to transition to the slow clock, the edge of the slow clock is detected and this causes the fast clock value to park. At this point the fast clock can be gated. On the next rising edge of the slow clock, the parked fast clock value (in the upper 64 bits of an 82 bit value) is added to the value from the Off-load Calibrator. On subsequent edges while in slow clock mode the slow clock counter increments its count by the value from the Off-load Calibrator.

When transitioning from slow clock to fast clock, the fast clock waits until it samples a rising edge of the slow clock through its synchronizer and then loads the upper 64 bits of the slow clock value as the fast count value. It then de-asserts the indication that slow clock mode is active. The 32 kHz clock counter no longer counts. The 64 bit MSB will be over-written when the 32 kHz counter is reloaded once conditions are met to enable the 32 kHz HPET counter but the 18 bit LSB is retained and it is not cleared out during the next reload cycle to avoid losing the fractional part of the counter.

After initiating a transition from fast clock to slow clock and parking the fast counter value, the fast counter no longer tracks. This means if a transition back to fast clock is requested before the entry into off-load slow clock mode completes, the Off-loadable Timer must wait until the next slow clock edge to restart. This case effectively performs the fast clock to slow clock and back to fast clock on the same slow clock edge.

24.2.3.3 Periodic Versus Non-Periodic Modes

Non-Periodic Mode

This mode can be thought of as creating a one-shot timer.

When a timer is set up for non-periodic mode, it will generate an interrupt when the value in the main counter matches the value in the timer’s comparator register. Another interrupt will be generated when the main counter matches the value in the timer’s comparator register after a wrap around.

During run-time, the value in the timer’s comparator value register will not be changed by the hardware. Software can of course change the value.

The Timer 0 Comparator Value register cannot be programmed reliably by a single 64 bit write in a 32 bit environment except if only the periodic rate is being changed during run-time. If the actual Timer 0 Comparator Value needs to be reinitialized, then the following software solution will always work regardless of the environment:

- Set TIMER0_VAL_SET_CNF bit
- Set the lower 32 bits of the Timer0 Comparator Value register
- Set TIMER0_VAL_SET_CNF bit
- Set the upper 32 bits of the Timer0 Comparator Value register

Timer 0 is configurable to 32 (default) or 64 bit mode, whereas Timers 1:7 only support 32 bit mode.

WARNING

Software must be careful when programming the comparator registers. If the value written to the register is not sufficiently far in the future, then the counter may pass the value before it reaches the register and the interrupt will be missed. The BIOS should pass a data structure to the operating system to indicate that the operating system should not attempt to program the periodic timer to a rate faster than 5 us.

All of the timers support non-periodic mode.

Refer to *IA-PC HPET Specification* for more details of this mode.

Periodic Mode

When a timer is set up for periodic mode, the software writes a value in the timer's comparator value register. When the main counter value matches the value in the timer's comparator value register, an interrupt can be generated. The hardware will then automatically increase the value in the comparator value register by the last value written to that register.

To make the periodic mode work properly, the main counter is typically written with a value of 0 so that the first interrupt occurs at the right point for the comparator. If the main counter is not set to 0, interrupts may not occur as expected.

During run-time, the value in the timer's comparator value register can be read by software to find out when the next periodic interrupt will be generated (not the rate at which it generates interrupts). Software is expected to remember the last value written to the comparator's value register (the rate at which interrupts are generated).

If software wants to change the periodic rate, it should write a new value to the comparator value register. At the point when the timer's comparator indicates a match, this new value will be added to derive the next matching point.

If the software resets the main counter, the value in the comparator's value register needs to reset as well. This can be done by setting the TIMERn_VAL_SET_CNF bit. Again, to avoid race conditions, this should be done with the main counter halted. The following usage model is expected:

1. Software clears the ENABLE_CNF bit to prevent any interrupts.
2. Software Clears the main counter by writing a value of 00h to it.
3. Software sets the TIMER0_VAL_SET_CNF bit.
4. Software writes the new value in the TIMER0_COMPARATOR_VAL register.

Software sets the ENABLE_CNF bit to enable interrupts.

NOTE

As the timer period approaches zero, the interrupts associated with the periodic timer may not get completely serviced before the next timer match occurs. Interrupts may get lost and/or system performance may be degraded in this case.

Each timer is NOT required to support the periodic mode of operation. A capabilities bit indicates if the particular timer supports periodic mode. The reason for this is that supporting the periodic mode adds a significant amount of gates.

Only timer 0 will support the periodic mode. This saves a substantial number of gates.

24.2.3.4 Enabling the Timers

The BIOS or operating system PnP code should route the interrupts. This includes the Legacy Rout bit, Interrupt Rout bit (for each timer), and interrupt type (to select the edge or level type for each timer).

The Device Driver code should do the following for an available timer:

1. Set the Overall Enable bit (Offset 10h, bit 0).
2. Set the timer type field (selects one-shot or periodic).
3. Set the interrupt enable.
4. Set the comparator value.

24.2.3.5 Interrupt Levels

Interrupts directed to the internal 8259s are active high. Refer to the **Advanced Programmable Interrupt Controller (APIC) (D31:F0)** for information regarding the polarity programming of the I/O APIC for detecting internal interrupts.

If the interrupts are mapped to the 8259 or I/O APIC and set for level-triggered mode, they can be shared with legacy interrupts. They may be shared although it is unlikely for the operating system to attempt to do this.

If more than one timer is configured to share the same IRQ (using the TIMERn_INT_ROUT_CNF fields), then the software must configure the timers to level-triggered mode. Edge-triggered interrupts cannot be shared.

For handling interrupts and issues related to 64 bit timers with 32 bit processors, refer to IA-PC HPET Specification.

25.0 Intel® Serial IO Inter-Integrated Circuit (I2C) Controllers

The Processor implements six I^2C controllers for six independent I^2C interfaces, I2C0-I2C5. Each interface is a two-wire serial interface consisting of a serial data line (SDA) and a serial clock (SCL).

I2C4 and I2C5 only implement the I^2C host controllers and do not incorporate a DMA controller. Therefore, I2C4 and I2C5 are restricted to operate in PIO mode only.

The I^2C interfaces support the following features:

- Speed: standard mode (up to 100 Kb/s), fast mode (up to 400 Kb/s), fast mode plus (up to 1 Mb/s) and High speed mode (up to 3.2 Mb/s) and I3C mode.
- Operate in 1.8 V only
- Host I^2C operation only
- 7-bit or 10-bit addressing
- 7-bit or 10-bit combined format transfers
- Bulk transmit mode
- Ignoring CBUS addresses (an older ancestor of I^2C used to share the I^2C bus)
- Interrupt or polled-mode operation
- Bit and byte waiting at all bus speed
- Component parameters for configurable software driver support
- Programmable SDA hold time (tHD; DAT)
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- 64-byte Tx FIFO and 64-byte Rx FIFO
- SW controlled serial data line (SDA) and serial clock (SCL)

NOTES

1. The controllers must only be programmed to operate in Host mode only. I^2C device mode is not supported.
2. I^2C multi hosts is not supported.
3. Simultaneous configuration of Fast Mode and Fast Mode Plus/High speed mode is not supported.
4. I^2C General Call is not supported.

Table 82. Acronyms

Acronyms	Description
I^2C	Inter-Integrated Circuit
PIO	Programmed Input/Output
SCL	Serial Clock Line
SDA	Serial Data Line

Table 83. References

Specification	Location
The I^2C Bus Specification, Version 5	www.nxp.com/documents/user_manual/UM10204.pdf

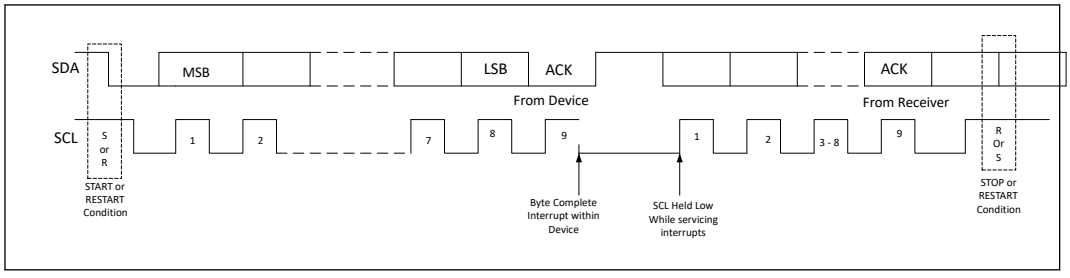
25.1 Functional Description

25.1.1 Protocols Overview

For more information on the I^2C protocols and command formats, refer to the industry I^2C specification. Below is a simplified description of I^2C bus operation:

- The host generates a START condition, signaling all devices on the bus to listen for data.
- The host writes a 7-bit address, followed by a read/write bit to select the target device and to define whether it is a transmitter or a receiver.
- The target device sends an acknowledge bit over the bus. The host must read this bit to determine whether the addressed target device is on the bus.
- Depending on the value of the read/write bit, any number of 8-bit messages can be transmitted or received by the host. These messages are specific to the I^2C device used. After 8 message bits are written to the bus, the transmitter will receive an acknowledge bit. This message and acknowledge transfer continues until the entire message is transmitted.
- The message is terminated by the host with a STOP condition. This frees the bus for the next host to begin communications. When the bus is free, both data and clock lines are high.

Figure 15. Data Transfer on I^2C Bus



Combined Formats

The Processor I^2C controllers support mixed read and write combined format transactions in both 7-bit and 10-bit addressing modes.

The Processor controllers do not support mixed address and mixed address format (which means a 7-bit address transaction followed by a 10-bit address transaction or vice versa) combined format transaction.

To initiate combined format transfers, IC_CON.IC_RESTSART_EN should be set to 1. With this value set and operating as a host, when the controller completes an I^2C transfer, it checks the transmit FIFO and executes the next transfer. If the direction of this transfer differs from the previous transfer, the combined format is used to issue the transfer. If the transmit FIFO is empty when the current I^2C transfer completes, a STOP is issued and the next transfer is issued following a START condition.

25.1.2 DMA Controller

The I^2C controllers 0 to 3 (I2C0 - I2C3) each has an integrated DMA controller. The I^2C controller 4 and 5 (I2C4 and I2C5) only implement the I^2C host controllers and do not incorporate a DMA. Therefore, I2C4 and I2C5 are restricted to operate in PIO mode only.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires the peripheral to control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires the peripheral to control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory (such as DDR or SRAM). The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode.

Channel Control

- The source transfer width and destination transfer width is programmable. The width can be programmed to 1, 2, or 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. This number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. Block size is not limited by the source or destination transfer widths.
- Address incrementing modes: The DMA has a configurable mechanism for computing the source and destination addresses for the next transfer within the current block. The DMA supports incrementing addresses and constant addresses.
- Flexibility to configure any hardware handshake sideband interface to any of the DMA channels.

- Early termination of a transfer on a particular channel.

25.1.3 Reset

Each host controller has an independent reset associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

NOTE

To avoid a potential I^2C peripheral deadlock condition where the reset goes active in the middle of a transaction, the I^2C controller must be idle before a reset can be initiated.

25.1.4 Power Management

Device Power Down Support

To power down peripherals connected to Processor I^2C bus, the idle configured state of the I/O signals is retained to avoid voltage transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when I^2C bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The interface supports this by reporting its service latency requirements to the platform power management controller using LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

25.1.5 Interrupts

I^2C interface has an interrupt line which is used to notify the driver that service is required.

When an interrupt occurs, the device driver needs to read the host controller, DMA interrupt status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

25.1.6 Error Handling

Errors that might occur on the external I^2C signals are comprehended by the I^2C host controller and reported to the I^2C bus driver through the MMIO registers.

25.1.7 Programmable SDA Hold Time

The Processor includes a software programmable register to enable dynamic adjustment of the SDA hold time, if needed.

25.2 Signal Description

Signal Name	Type	Description
GPP_H19/**I2C0_SDA**/I3C0_SDA	I/OD	**I^2C Link 0 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H20/**I2C0_SCL**/I3C0_SCL	I/OD	**I^2C Link 0 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H21/**I2C1_SDA**/I3C1_SDA	I/OD	**I^2C Link 1 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H22/**I2C1_SCL**/I3C1_SCL	I/OD	**I^2C Link 1 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H04/**I2C2_SDA**/ CNV_MFUART2_RXD	I/OD	**I^2C Link 2 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H05/**I2C2_SCL**/ CNV_MFUART2_TXD	I/OD	**I^2C Link 2 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B02/ISH_I2C0_SDA/ ISH_I3C0_SDA/**I2C2A_SDA**	I/OD	**I^2C Link 2A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C2 controller, to support touch device interface convergence.
GPP_B03/ISH_I2C0_SCL/ ISH_I3C0_SCL/**I2C2A_SCL**	I/OD	**I^2C Link 2A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C2 controller, to support touch device interface convergence.
GPP_H06/**I2C3_SDA**/UART1_RXD/ ISH_UART1A_RXD	I/OD	**I^2C Link 3 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H07/**I2C3_SCL**/UART1_TXD/ ISH_UART1A_TXD	I/OD	**I^2C Link 3 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_D01/**I2C3A_SDA**/ ISH_I2C2A_SDA	I/OD	**I^2C Link 3A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C3 controller, to support touch device interface convergence.
GPP_D02/**I2C3A_SCL**/ ISH_I2C2A_SCL	I/OD	**I^2C Link 3A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.

continued...

Signal Name	Type	Description
		Note : Alternate interface from/to the same I2C3 controller, to support touch device interface convergence.
GPP_E12/THC_I2C0_SCL/ THC0_SPI1_IO0/GSPI0_MOSI/ **I2C4_SCL**	I/OD	**I²C Link 4 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_E13/THC_I2C0_SDA/ THC0_SPI1_IO1/GSPI0_MISO/ **I2C4_SDA**	I/OD	**I²C Link 4 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B18/ISH_I2C2_SDA/**I2C4A_SDA**	I/OD	**I²C Link 4A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C4 controller, to support touch device interface convergence.
GPP_B19/ISH_I2C2_SCL/**I2C4A_SCL**	I/OD	**I²C Link 4A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C4 controller, to support touch device interface convergence.
GPP_F13/THC_I2C1_SDA/I3C2_SDA/ THC1_SPI2_IO1/ISH_SPIA_MOSI/ GSPI1_MISO/**I2C5_SDA**	I/OD	**I²C Link 5 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_F12/THC_I2C1_SCL/I3C2_SCL/ THC1_SPI2_IO0/ISH_SPIA_MISO/ GSPI1_MOSI/**I2C5_SCL**	I/OD	**I²C Link 5 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B20/**I2C5A_SDA**/ISH_GP8	I/OD	**I²C Link 5A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C5 controller, to support touch device interface convergence.
GPP_B21/**I2C5A_SCL**/ISH_GP9	I/OD	**I²C Link 5A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C5 controller, to support touch device interface convergence.

25.3 Integrated Pull-Ups and Pull-Downs

None.

25.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
I2C[5:0]_SDA , I2C[2:5]A_SDA	Primary	Undriven	Undriven	Undriven
I2C[5:0]_SCL , I2C[2:5]A_SCL	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

26.0 Intel® Serial IO Improved Inter-Integrated Circuit (I3C) Controllers

I^3C specification is backward compatible with I^2C devices. The I^3C enables dynamic address allocation and inband interrupts. The Spec also allows for hot-plug / hot-join of devices. The I^3C Specification is backward compatible with legacy I^2C devices and enables coexistence of legacy I^2C and I^3C devices on the same bus in Fast Mode, Fast Mode Plus modes, without clock stretching. The processor has two I^3C controller compliant to MIPI I^3C HCI Specification, that can support 3 I^3C buses and up to 8 devices per bus (subject to meeting electrical/topology requirements).

The I^3C interfaces support the following features:

- Support for MIPI I^3C spec v1.0, and MIPI I^3C HCI Specification.
- Speed: standard mode (up to 100 Kb/s), fast mode (up to 400 Kb/s), fast mode plus (up to 1 MB/s) and I3C Mode.
- Support clock loopback using dummy IO to meet ACIO timing.
- Maximum theoretical Baud rate is 12900 kbps
- Maximum validated Baud rate is 12500 kbps
- Operate in 1.8 V Only
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- 64-byte Tx FIFO and 64-byte Rx FIFO
- PME/wake support for IBI, when in S0ix
- PCI/ACPI enumeration support
- I^3C static addressing and dynamic addressing support
- I^3C in-band interrupt
- I^3C transactions using SDR
- Error detection and recovery methods M0, M2
- For stalling Host clock on data buffering
- Host I^3C operation only

NOTES

1. The controllers must only be programmed to operate in Host mode only. I^3C Device mode is not supported.
2. I^3C multi-host Mode is not supported.
3. Simultaneous configuration of Fast Mode and Fast Mode Plus is not supported.

Table 84. Acronyms

Acronyms	Description
I^3C	Improved Inter-Integrated Circuit
SCL	Serial Clock Line
SDA	Serial Data Line

26.1 Functional Description

26.1.1 Reset

Each host controller has an independent reset associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

NOTE

To avoid a potential I^3C peripheral deadlock condition where the reset goes active in the middle of a transaction, the I^3C controller must be idle before a reset can be initiated.

26.1.2 Power Management

Device Power Down Support

To power down peripherals connected to Processor I^3C bus, the idle configured state of the I/O signals is retained to avoid voltage transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when I^3C bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The interface supports this by reporting its service latency requirements to the platform power management controller using LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements.

2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

26.1.3 Interrupts

I^3C interface has an interrupt line which is used to notify the driver that service is required.

When an interrupt occurs, the device driver needs to read the host controller, DMA interrupt status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

26.2 Signal Description

Signal Name	Type	Description
GPP_H19/I2C0_SDA/**I3C0_SDA**	I/OD	**I^3C Link 0 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H20/I2C0_SCL/**I3C0_SCL**	I/OD	**I^3C Link 0 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H21/I2C1_SDA/**I3C1_SDA**	I/OD	**I^3C Link 1 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H22/I2C1_SCL/**I3C1_SCL**	I/OD	**I^3C Link 1 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H10/UART0_RTS#/**I3C1A_SDA**/ ISH_GP10A	I/OD	**I^3C Link 1A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I3C1 controller, to support touch device interface convergence.
GPP_H11/UART0_CTS#/**I3C1A_SCL**/ ISH_GP11A	I/OD	**I^3C Link 1A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I3C1 controller, to support touch device interface convergence.
GPP_F13/THC_I2C1_SDA/**I3C2_SDA**/ THC1_SPI2_IO1/ISH_SPIA_MOSI/ GSPI1_MISO/I2C5_SDA	I/OD	**I^3C Link 2 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_F12/THC_I2C1_SCL/**I3C2_SCL**/ THC1_SPI2_IO0/ISH_SPIA_MISO/ GSPI1_MOSI/I2C5_SCL	I/OD	**I^3C Link 2 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.

26.3 Integrated Pull-Ups and Pull-Downs

Signals	Resistor Type	Value	Notes
I3C [2:0]_SDA, I3C1A_SDA	Pull-Up	5 kohm	
I3C [2:0]_SCL, I3C1A_SCL	Pull-Up	5 kohm	

26.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
I3C[2:0]_SDA , I3C1A_SDA	Primary	Undriven	Undriven	Undriven
I3C[2:0]_SCL, I3C1A_SCL	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

27.0 Gigabit Ethernet Controller

The Gigabit Ethernet controller in conjunction with the Intel® Ethernet Connection I219 provides a complete LAN solution. This chapter describes the behavior of the Gigabit Ethernet Controller. For details on the Intel® Ethernet Connection I219, refer to Intel® Ethernet Connection I219 Datasheet (#544486).

Table 85. Acronyms

Acronyms	Description
GbE	Gigabit Ethernet

Table 86. References

Specification	Location
IEEE 802.3 Ethernet	http://standards.ieee.org/getieee802/
Intel® Ethernet Connection I219 Datasheet	http://www.intel.com/content/www/us/en/

27.1 Functional Description

The processor integrates a Gigabit Ethernet (GbE) controller. The integrated GbE controller is compatible with the Intel® Ethernet Connection I219. The integrated GbE controller provides two interfaces for 10/100/1000 Mbps and manageability operation:

- Data link based on PCI Express* – A high-speed interface that uses PCIe* electrical signaling at half speed and custom logical protocol for active state operation mode.
- System Management Link (SMLink0)—A low speed connection for low power state mode for manageability communication only. The frequency of this connection can be configured to one of three different speeds (100 kHz, 400 kHz, or 1 MHz).

The Intel® Ethernet Connection I219 only runs at a speed of 1250 Mbps, which is 1/2 of the 2.5 GB/s PCI Express* frequency. Some of the PCI Express* root ports in the processor have the ability to run at the 1250-Mbps rate. There is no need to implement a mechanism to detect that the Platform LAN Device is connected. The port configuration (if any), attached to the Platform LAN Device, is pre-loaded from the NVM. The selected port adjusts the transmitter to run at the 1250-Mbps rate and does not need to be PCI Express* compliant.

NOTE

PCIe* validation tools cannot be used for electrical validation of this interface—however, PCIe* layout rules apply for on-board routing.

NOTE

Refer the section "Flexible High Speed I/O" for GbE lane allocation options.

The integrated GbE controller operates at full-duplex at all supported speeds or half-duplex at 10/100 Mbps. It also adheres to the *IEEE 802.3x Flow Control Specification.*

NOTE

GbE operation (1000 Mbps) is only supported in S0 mode. In Sx modes, the platform LAN Device may maintain 10/100 Mbps connectivity and use the SMLink interface to communicate with the processor.

The integrated GbE controller provides a system interface using a PCI function. A full memory-mapped or I/O-mapped interface is provided to the software, along with DMA mechanisms for high performance data transfer.

The integrated GbE controller features are:

- Network Features
  - Compliant with the 1 GB/s Ethernet 802.3, 802.3u, 802.3ab specifications
  - Multi-speed operation: 10/100/1000 Mbps
  - Full-duplex operation at 10/100/1000 Mbps: Half-duplex at 10/100 Mbps
  - Flow control support compliant with the 802.3X specification
  - VLAN support compliant with the 802.3q specification
  - MAC address filters: perfect match unicast filters; multicast hash filtering, broadcast filter and promiscuous mode
  - PCI Express*/SMLink interface to GbE PHYs
- Host Interface Features
  - 64-bit address host support for systems using more than 4 GB of physical memory
  - Programmable host memory receive buffers (256 bytes to 16 KB)
  - Intelligent interrupt generation features to enhance driver performance
  - Descriptor ring management hardware for transmit and receive
  - Software controlled reset (resets everything except the configuration space)
  - Message Signaled Interrupts
- Performance Features
  - Configurable receive and transmit data FIFO, programmable in 1 KB increments
  - TCP segmentation off loading features
  - Fragmented UDP checksum off load for packet reassembly
  - IPv4 and IPv6 checksum off load support (receive, transmit, and large send)
  - Split header support to eliminate payload copy from user space to host space
  - Receive Side Scaling (RSS) with two hardware receive queues
  - Supports 9018 bytes of jumbo packets
  - Packet buffer size 32 KB
  - TimeSync off load compliant with 802.1as specification
  - Platform time synchronization

- Power Management Features
  - Magic Packet* wake-up enable with unique MAC address
  - ACPI register set and power down functionality supporting D0 and D3 states
  - Full wake up support (APM, ACPI)
  - MAC power down at Sx, DM-Off with and without WoL
  - Auto connect battery saver at S0 no link and Sx no link
  - Energy Efficient Ethernet (EEE) support
  - Latency Tolerance Reporting (LTR)
  - ARP and ND proxy support through LAN Connected Device proxy

27.1.1 GbE PCI Bus Interface

The GbE controller has a PCI interface to the host processor and host memory. The following sections detail the bus transactions.

Transaction Layer

The upper layer of the host architecture is the transaction layer. The transaction layer connects to the device GbE controller using an implementation specific protocol. Through this GbE controller-to-transaction-layer protocol, the application-specific parts of the device interact with the subsystem and transmit and receive requests to or from the remote agent, respectively.

Data Alignment

- **4-KB Boundary**

  PCI requests must never specify an address/length combination that causes a memory space access to cross a 4-KB boundary. It is hardware's responsibility to break requests into 4-KB aligned requests (if needed). This does not pose any requirement on software. However, if software allocates a buffer across a 4-KB boundary, hardware issues multiple requests for the buffer. Software should consider aligning buffers to a 4-KB boundary in cases where it improves performance. The alignment to the 4-KB boundaries is done by the GbE controller. The transaction layer does not do any alignment according to these boundaries.

- **PCI Request Size**

  PCI requests are 64 bytes or less and are aligned to make better use of memory controller resources.

Configuration Request Retry Status

The integrated GbE controller might have a delay in initialization due to an NVM read. If the NVM configuration read operation is not completed and the device receives a configuration request, the device responds with a configuration request retry completion status to terminate the request, and thus effectively stalls the configuration request until such time that the sub-system has completed local initialization and is ready to communicate with the host.

27.1.2 Error Events and Error Reporting

Complete Abort Error Handling

A received request that violates the LAN Controller programing model will be discarded, for non posted transactions an unsuccessful completion with CA completion status will be returned.

Unsupported Request Error Handling

A received unsupported request to the LAN Controller will be discarded, for non posted transactions an unsuccessful completion with UR completion status will be returned. The URD bit will be set in ECTL register.

27.1.3 Ethernet Interface

The integrated GbE controller provides a complete CSMA/CD function supporting IEEE 802.3 (10 Mbps), 802.3u (100 Mbps) implementations. It also supports the IEEE 802.3z and 802.3ab (1000 Mbps) implementations. The device performs all of the functions required for transmission, reception, and collision handling called out in the standards.

The mode used to communicate between the processor and the Intel® Ethernet Connection I219 supports 10/100/1000 Mbps operation, with both half- and full-duplex operation at 10/100 Mbps, and full-duplex operation at 1000 Mbps.

Intel® Ethernet Connection I219

The integrated GbE controller and the Intel® Ethernet Connection I219 communicate through the PCIe* and SMLink0 interfaces. All integrated GbE controller configuration is performed using device control registers mapped into system memory or I/O space. The Platform LAN Phy is configured using the PCI Express or SMLink0 interface.

The integrated GbE controller supports various modes as listed in below table.

Table 87. LAN Mode Support

Mode	System State	Interface Active	Connections
Normal 10/100/1000 Mbps	S0	PCI Express*	Intel® Ethernet Connection I219
Normal 10/100/1000 Mbps	S0ix	SMLink0	
Wake-on-LAN	S0ix / Sx	SMLink0 / Wake#	
Manageability	S0ix / Sx	SMLink0	

27.1.4 PCI Power Management

The integrated GbE controller supports the Advanced Configuration and Power Interface (ACPI) specification as well as Advanced Power Management (APM). This enables the network-related activity (using an internal host wake signal) to wake up the host. For example, from S4 to S0.

The integrated GbE controller contains power management registers for PCI and supports D0 and D3 states. PCI transactions are only allowed in the D0 state, except for host accesses to the integrated GbE controller's PCI configuration registers.

The processor controls the voltage rails into the external LAN PHY using the SLP_LAN# pin.

- The LAN PHY is always powered when the Host and Intel® CSME systems are running.
  - SLP_LAN#='1' whenever SLP_S3#='1' or SLP_A#='1'.
- If the LAN PHY is required by Intel® CSME in Sx/M-Off , Intel® CSME must configure SLP_LAN#='1' irrespective of the power source and the destination power state. Intel® CSME must be powered at least once after G3 to configure this.
- If the LAN PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN.
- If the LAN PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN.
- If the LAN PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS.

27.2 Signal Description

Table 88. GbE LAN Signals

Signal Name	Type	Description	Availability
PCIE_C2_TX_P/**GbE_TX_P** PCIE_C2_TX_N/**GbE_TX_N**	O	Differential transmit pairs to the Intel® Ethernet Connection I219 based on the PCIe interface.	All Processor Series
PCIE_C2_RX_P/**GbE_RX_P** PCIE_C2_RX_N/**GbE_RX_N**	I	Differential receive pairs to the Intel® Ethernet Connection I219 based on the PCIe interface.	All Processor Series
GPP_C04/**SML0DATA**	I/OD	System Management Link data signal interface to Intel® Ethernet Connection I219. *Note:* The Intel® Ethernet Connection I219 connects to SML0DATA signal.	All Processor Series
GPP_C03/**SML0CLK**	I/OD	System Management Link data signal interface to Intel® Ethernet Connection I219. *Note:* The Intel® Ethernet Connection I219 connects to SML0CLK signal.	All Processor Series
GPP_V10/**LANPHYPC**	O	**LAN PHY Power Control:** LANPHYPC should be connected to LAN_DISABLE_N on the PHY. Processor will drive LANPHYPC low to put the PHY into a low power state when functionality is not needed. *Note:* LANPHYPC can only be driven low if SLP_LAN# is de-asserted.	All Processor Series
GPP_V11/**SLP_LAN#**	IO	**LAN Sub-System Sleep Control:** If the Gigabit Ethernet Controller is enabled, when SLP_LAN# is de-asserted it indicates that the PHY device must be powered. When SLP_LAN# is asserted, power can be shut off to the PHY device.	All Processor Series

continued...

Signal Name	Type	Description	Availability
		Note: If Gigabit Ethernet Controller is statically disabled via BIOS, SLP_LAN# will be driven low.	
GPP_V02/**SOC_WAKE#**	I	**SOC_WAKE:** LAN Wake Indicator from the GbE PHY. *Note:* SOC_WAKE# functionality is only supported with Intel PHY I219. Connection of a third party LAN device's wake signal to SOC_WAKE# is not supported.	All Processor Series

27.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SOC_WAKE#	External Pull-up required.	4.7 kohm +/- 5%	10 kohm +/- 5% pull-up resistor is also acceptable.

27.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S4/S5
SOC_WAKE#	Primary	Undriven	Undriven	Undriven[1]
SLP_LAN#	Primary	0	0	0/1[2]

Notes: 1. Based on wake events and Intel CSME state
2. Configurable based on BIOS settings: '0' When LAN controller is configured as "Disabled" in BIOS, SLP_LAN# will drive "Low";'1' When LAN controller is configured as "Enabled" in BIOS, SLP_LAN# will drive "High"

28.0 Connectivity Integrated (CNVi)

Connectivity Integrated (CNVi) is a general term referring to a family of connectivity solutions which are based on the Connectivity Controller family. The common component of all these solutions is the Connectivity Controller, which is embedded in the processor.

The Integrated Connectivity (CNVi) solution consists of the following entities:

- The containing chip (the processor which contains the Connectivity Controller)
- Buttress (as applicable to each platform, and coupled the Connectivity Controller)
- Companion RF chip that is in a pre-certified module (i.e., M.2-2230, M.2-1216) or soldered as chip on board.

The main blocks of the integrated Connectivity solution are partitioned according to the following:

Table 89. Acronyms

Acronyms	Description
BRI	Bluetooth* Radio Interface
CNVi	Connectivity Integrated
SCU	System Controller Unit . It is the controller unit of CNVi.
RGI	Radio Generic interface
IP	Literally, Intellectual Property. IP refers to architecture, design, validation, and software components collectively delivered to enable one or more specific to the processor features.
MFUART	Multifunction Universal Asynchronous Receiver/Transmitter
UART	Universal Asynchronous Receiver/Transmitter

28.1 Functional Description

The main blocks of the integrated Connectivity solution are partitioned according to the following:

- **Connectivity Controller IP** contains:
  - Controller IP supports both CNVio2 and CNVio3 also called as STEP (Serial Time Encoded Protocol) interface for Wi-Fi* 7 support. The CNVio3 uses the same clock signals of CNVio2 and does not use the CNVio2 data signals.
  - Interfaces to the processor
  - Debug and testing interfaces
  - Power management and clock Interfaces
  - Interface to the Companion RF module (CRF)
  - Interface to physical I/O pins controlled by the processor.
  - Interfaces to the LTE modem via processor GPIO

- **Companion RF (CRF)**: This is the integrated connectivity M.2 module. The CRF Top contains:
  - Debug and testing interfaces
  - Power and clock Interfaces
  - Interface to the Connectivity Controller chip
- **Physical I/O Pins**: The SCU units are responsible for generating and controlling the power and clock resources of Connectivity Controller and CRF. There are unique SCUs in Connectivity Controller and CRF and their operation is coordinated due to power and clock dependencies. This coordination is achieved by signaling over a control bus (AUX) connecting Connectivity Controller and CRF.

Both Connectivity Controller and CRF have a dedicated AUX bus and arbiter. These two AUX buses are connected by a special interface that connects over the RGI bus. Each of the Connectivity Controller and CRF cores is dedicated to handle a specific connectivity function (Wi-Fi, Bluetooth).

Only the digital part of the connectivity function is located in Connectivity Controller cores, while the CRF cores handle some digital, but mostly analog and RF functionality. Each core in the Connectivity Controller has an interface to the host and an interface to its counterpart in CRF. CRF cores include an analog part which is connected to board level RF circuitry and to an antenna.

28.2 Signal Description

Signal Name	Type	Description
GPIO fixed functions (Signals for Integrated Connectivity (CNVi) and Discrete Connectivity (CNVd) functions		
GPP_S04/SNDW2_CLK/DMIC_CLK_A0/ **I2S2_SCLK**	I/O	For CNVi: Unused For discrete connectivity with UART host support: Optional Bluetooth* I^2S bus clock
GPP_F04/**CNV_RF_RESET#**	I/O	For CNVi: RF companion (CRF) reset signal, active low. Require a 75 kohm Pull-Down on platform/motherboard level. It is recommended not to use it for bootstrapping during early Platform init flows.
GPP_S06/SNDW2_DATA1/SNDW1_CLK/ DMIC_CLK_A1/**I2S2_TXD**	O	For CNVi: Unused For discrete connectivity with UART host Bluetooth* support: Optional Bluetooth* I^2S bus data output (input to Bluetooth* module)
GPP_S07/SNDW3_DATA3/SNDW2_DATA2/ SNDW1_DATA0/DMIC_DATA1/**I2S2_RXD**	I	For CNVi: Unused. For discrete connectivity with UART host support: Optional Bluetooth* I^2S bus data output (input to Bluetooth* module)
GPP_F00/**CNV_BRI_DT**/UART2_RTS#	O	For CNVi: BRI bus TX. For discrete connectivity with UART host support: Bluetooth* UART RTS#
GPP_F01/**CNV_BRI_RSP**/UART2_RXD	I	For CNVi: BRI bus RX. For discrete connectivity with UART host support: Bluetooth* UART RXD
GPP_F02/**CNV_RGI_DT**/UART2_TXD	O	For CNVi: RGI bus TX. RGI_DT is used by the platform to strap presence of the CRF. Requires weak pull up of 20Kohm on the platform. For discrete connectivity with UART host support: Bluetooth* UART TXD
GPP_F03/**CNV_RGI_RSP**/UART2_CTS#	I	For CNVi: RGI bus RX.

continued...

Signal Name	Type	Description
		For discrete connectivity with UART host support: Bluetooth* UART CTS#
GPP_F05/**CRF_CLKREQ**	O	For CNVi: Processor to CRF wake indication
GPP_F06/**CNV_PA_BLANKING**	I/O	For CNVi and discrete connectivity : Optional WLAN/Bluetooth* WWAN co-existence signal. Used to be co-existence signal for external GNSS solution
GPP_H04/I2C2_SDA/**CNV_MFUART2_RXD**	I	For CNVi and discrete connectivity: Optional WLAN/Bluetooth* WWAN co-existence signal (Input)
GPP_H05/I2C2_SCL/**CNV_MFUART2_TXD**	O	For CNVi and discrete connectivity : Optional WLAN/Bluetooth* WWAN co-existence signal (Output)
Fixed special purpose I/O		
CNV_WT_CLKP	O	CNVio bus TX CLK+
CNV_WT_CLKN	O	CNVio bus TX CLK-
CNV_WT_D0P	O	CNVio bus Lane 0 TX+
CNV_WT_D0N	O	CNVio bus Lane 0 TX-
CNV_WT_D1P	O	CNVio bus Lane 1 TX+
CNV_WT_D1N	O	CNVio bus Lane 1 TX-
CNV_WR_CLKP	I	CNVio bus RX CLK+
CNV_WR_CLKN	I	CNVio bus RX CLK-
CNV_WR_D0P	I	CNVio bus Lane 0 RX+
CNV_WR_D0N	I	CNVio bus Lane 0 RX-
CNV_WR_D1P	I	CNVio bus Lane 1 RX+
CNV_WR_D1N	I	CNVio bus Lane 1 RX-
Selectable special purpose I/O		
USB2P_8	I/O	Bluetooth* USB host bus (positive) for discrete connectivity. Optional to connect to a Bluetooth* USB+ pin on the Bluetooth* module. Other USB 2.0 ports can be selected for this function.
USB2N_8	I/O	Bluetooth* USB host bus (negative) for discrete connectivity. Optional to connect to a Bluetooth* USB- pin on the Bluetooth* module. Other USB 2.0 ports can be selected for this function.
PCIE_A4_TX_P	O	Wi-Fi* PCIe* host bus TX (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PERp0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_A4_TX_N	O	Wi-Fi* PCIe* host bus TX (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PERn0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_A4_RX_P	I	Wi-Fi* PCIe* host bus RX (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PETp0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_A4_RX_N	I	Wi-Fi* PCIe* host bus RX (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PETn0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
CLKOUT_P4	O	Wi-Fi* PCIe* host bus clock (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* REFCLKp pin on the Wi-Fi* module. Other PCIe* clocks can be selected for this function.

continued...

Signal Name	Type	Description
CLKOUT_N4	O	Wi-Fi* PCIe* host bus clock (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* REFCLKp pin on the Wi-Fi* module. Other PCIe* clocks can be selected for this function.
CL_RST#	O	Wi-Fi* CLINK host bus reset for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK reset pin on the Intel® vPro™ Wi-Fi* module.
CL_DATA	I/O	Wi-Fi* CLINK host bus data for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK data pin on the Intel® vPro™ Wi-Fi* module.
CL_CLK	O	Wi-Fi* CLINK host bus clock for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK clock pin on the Intel® vPro™ Wi-Fi* module.
W_Disable1# (GPIO)	O	Used for Wi-Fi* RF-Kill control. This pin can be connected to a platform switch or to processor GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init). The signal must keep value in Sx state (configured in BIOS) *Note:* Signal name not available in processor ballmap. This is a representation of GPIO used as CNVi signal.
W_Disable2# (GPIO)	O	Used for Bluetooth* RF-Kill control. This pin can be connected to a platform switch or to processor GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init). The signal must keep value in Sx state (configured in BIOS) *Note:* Signal name not available in processor ballmap. This is a representation of GPIO used as CNVi signal.
CNV_RCOMP	Analog	CNVi RCOMP is analog connection point for an external bias resistor(200ohms) to ground.

28.3 Integrated Pull-ups and Pull-downs

Signal	Resistor	Value	Notes
CNV_BRI_RSP	Pull up	20 kohm	
CNV_RGI_RSP	Pull up	20 kohm	

28.4 IO Signal Planes and States

Signal Name	Power plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CNV_RF_RESET#	Primary	Undriven	Driven Low	Undriven
CRF_CLKREQ	Primary	Driven High	Driven High	Driven High
CNV_MFUART2_RXD	Primary	Undriven	Undriven	Undriven
CNV_MFUART2_TXD	Primary	Undriven	Undriven	Undriven
CNV_BRI_DT	Primary	Driven High	Driven High	Driven High
CNV_BRI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)
CNV_RGI_DT	Primary	Driven High	Driven High	Driven High

continued...

Signal Name	Power plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CNV_RGI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)
CNV_WT_CLKP	Primary	Undriven	Driven Low	Undriven
CNV_WT_CLKN	Primary	Undriven	Driven Low	Undriven
CNV_WT_D0P	Primary	Undriven	Driven Low	Driven High
CNV_WT_D0N	Primary	Undriven	Driven Low	Driven High
CNV_WT_D1P	Primary	Undriven	Driven Low	Driven High
CNV_WT_D1N	Primary	Undriven	Driven Low	Driven High
CNV_WR_CLKP	Primary	Undriven	Undriven	Powered (input)
CNV_WR_CLKN	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D0P	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D0N	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D1P	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D1N	Primary	Undriven	Undriven	Powered (input)
CNV_RCOMP	Primary	Undriven	Undriven	Driven High

Note: 1. Reset reference for primary well pins is RSMRST#.

29.0 Controller Link

The controller link is used to manage the wireless devices supporting Intel® CSME Technology. Controller Link will transmit data at 60.0 Mbps on the Controller Link interface. The Controller Link clock frequency is 30.0 MHz. The Controller Link interface voltage supported is 1.25 V nominal.

NOTE

Refer to WNIC product datasheets for supported data rate and clock.

Table 90. Acronyms

Acronyms	Description
CL	Controller Link
WLAN	Wireless Local Area Network
WNIC	Wireless Network Interface Card

29.1 Signal Description

Signal Name	Type	Description
CL_DATA	I/O	**Controller Link Data:** Bi-directional data that connects to a Wireless LAN Device supporting Intel® Active Management Technology.
CL_CLK	O	**Controller Link Clock**: Bi-directional clock that connects to a Wireless LAN Device supporting Intel® Active Management Technology.
CL_RST#	O	**Controller Link Reset:**Controller Link reset that connects to a Wireless LAN Device supporting Intel® Active Management Technology.

29.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value (Ohm)	Notes
CL_DATA	Pull-up Pull-down	31.25 100	IO Signal Planes and States on page 204
CL_CLK	Pull-up Pull-down	31.25 100	

29.3 IO Signal Planes and States

Signal Name	Power Plane	During Reset[3]	Immediately After Reset[3]	S4/S5
CL_DATA	Primary	Refer to Notes	Refer to Notes	Internal Pull-down
CL_CLK	Primary	Refer to Notes	Refer to Notes	Internal Pull-down
CL_RST#	Primary	Driven Low	Driven High	Driven High

Notes: 1. The Controller Link clock and data buffers use internal Pull-up or Pull-down resistors to drive a logical 1 or 0.
2. The terminated state is when the I/O buffer Pull-down is enabled.
3. Reset reference for primary well pins is RSMRST#.

29.4 External CL_RST Pin Driven Open drained Mode Support

The WLAN has transitioned to 1.8 V for external CL_RST# pin and the processor controller Link I/O buffer drives 1.8 V only on this pin.

30.0 Integrated Sensor Hub (ISH)

The Integrated Sensor Hub (ISH) serves as the connection point for many of the sensors on a platform. The ISH is designed with the goal of "Always On, Always Sensing" and it provides the following functions to support this goal:

- Acquisition/sampling of sensor data.
- The ability to combine data from individual sensors to create a more complex virtual sensor that can be directly used by the firmware/OS.
- Low power operation through clock and power gating of the ISH blocks together with the ability to manage the power state of the external sensors.
- The ability to operate independently when the host platform is in a low power state (S0ix only).
- Ability to provide sensor-related data to other subsystems within the Processor , such as the Intel® CSME.

The ISH consists of the following key components:

- A combined cache for instructions and data.
  - — ROM space intended for the bootloader.
  - — SRAM space for code and data.
- Interfaces to sensor peripherals (I^2C, I^3C, UART, SPI, GPIO).
- An interface to main memory.
- Out of Band signals for clock and wake-up control.
- Inter Process Communications to the Host and Intel® CSME.
- Part of the PCI tree on the host.

Table 91. Acronyms

Acronyms	Description
Intel® CSME	Intel® Converged Security and Management Engine
I^2C	Inter-Integrated Circuit
IPC	Inter Process Communication
SPI	Serial Peripheral Interface
ISH	Integrated Sensor Hub
PMU	Power Management Unit
SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver/Transmitter

Table 92. References

Specification	Location
I^2C Specification Version 6.0	http://www.nxp.com/docs/en/user-guide/UM10204.pdf

30.1 Features

30.1.1 ISH I2C Controllers

The ISH supports three I^2C controllers capable of operating at speeds up to 2.4 Mbps each. The I^2C controllers are completely independent of each other: they do not share any pins, memory spaces, or interrupts.

The ISH's I^2C host controllers share the same general specifications:

- Host I^2C operation
- Support for the following operating speeds:
  - Standard mode: 100 kbps
  - Fast Mode: 400 kbps
  - Fast Mode Plus: 1000 kbps
  - High Speed Mode: 2400 kbps
- Support for both 7-bit and 10-bit addressing formats on the I^2C bus
- FIFO of 64 bytes with programmable watermarks/thresholds

30.1.2 ISH I3C Controllers

The ISH supports one I^3C controllers, two-Wire I^3C serial interface which consists of a Serial Data Line (SDA) and a Serial Clock (SCL).

The following are the ISH's I^3C host controller specifications:

- Modes Supported
  - I^3C Controller (Single Controller) - other modes are not supported
- In-band-interrupt (IBI) buffer depth of 16 bytes for command and status
- Supports Data transfer to legacy I^2C devices
- Clock stalling support in Controller Mode
- Supports various Data rates, such as, FM, FM+, SDR, HDR/DDR Rate
- CRC/Parity Generation and Validation
- Support for broadcast and directed CCC transfers
- Detects arbitration loss due to incoming IBI and subsequently re-transmits the command.
- Use of Duty Cycle to achieve Lower Effective Speed for SDR transfers to work with slower I^3C

30.1.3 ISH UART Controller

The ISH has two UART ports, each comprised of a four-wire, bi-directional point-to-point connection between the ISH and a peripheral.

The UART has the following capabilities:

- Support for operating speeds up to 4 Mbps
- Support for auto flow control using the RTS#/CTS# signals
- 64-byte FIFO
- DMA support to allow direct transfer to the ISH local SRAM without intervention by the controller. This saves interrupts on packets that are longer than the FIFO or when there are back-to-back packets to send or receive.

30.1.4 ISH GSPI Controller

The ISH supports one SPI controller comprises of four-wired interface connecting the ISH to external sensor devices.

The SPI controller includes:

- Operate in Host mode only
- Single Chip Select
- Half Duplex operation only
- Programmable SPI clock frequency range with maximum rate of 24 Mbits/sec
- FIFO of 64 bytes with programmable thresholds
- Support Programmable character length (2 to 16 bits)

30.1.5 ISH GPIOs

The ISH supports eight dedicated GPIOs.

30.2 Functional Description

This section provides the information about ISH Micro-Controller, SRAM, PCI Host Interface, Power Domains and Management, ISH IPC and ISH Interrupt Handling via IOAPIC (Interrupt Controller).

30.2.1 ISH Micro-Controller

The ISH is operated by a micro-controller. This core provides localized sensor aggregation and data processing, thus off loading the processor and lowering overall platform average power. The core supports an in-built local APIC that receives messages from the IOAPIC. A local boot ROM with FW for initialization is also part of the core.

30.2.2 SRAM

The local SRAM is used for ISH FW code storage and to read/write operational data. The local SRAM block includes both the physical SRAM as well as the controller logic. The SRAM is a total of 640 KB organized into banks of 32 KB each and is 32-bit wide.

The SRAM is shared with Intel® CSME as shareable memory. To protect against memory errors, the SRAM includes ECC support. The ECC mechanism is able to detect multi-bit errors and correct for single bit errors. The ISH firmware has the ability to put unused SRAM banks into lower power states to reduce power consumption.

30.2.3 PCI Host Interface

The ISH provides access to PCI configuration space via a PCI Bridge. Type 0 Configuration Cycles from the host are directed to the PCI configuration space.

MMIO Space

A memory-mapped Base Address Register (BAR0) with a set of functional memory-mapped registers is accessible to the host via the Bridge. These registers are owned by the driver running on the Host OS.

The bridge also supports a second BAR (BAR1) that is an alias of the PCI Con figuration space. It is used only in ACPI mode (that is, when the PCI con figuration space is hidden).

DMA Controller

The DMA controller supports up to 64-bit addressing.

PCI Interrupts

The PCI bridge supports standard PCI interrupts, delivered using IRQx to the system IOAPIC and not using an MSI to the host Processor.

PCI Power Management

PME is not supported in ISH.

30.2.4 ISH IPC

The ISH has IPC channels for communication with the Host Processor and Intel® CSME. The functions supported by the ISH IPC block are listed below.

Function 1: Allows for messages and interrupts to be sent from an initiator (such as the ISH) and a target (such as the Intel® CSME). The supported initiator -> target flows using this mechanism are shown in the table below.

Table 93. IPC Initiator -> Target flows

Initiator	Target
ISH	Host processor
Host processor	ISH
ISH	Intel® CSME
Intel® CSME	ISH

Function 2: Provides status registers and remap registers that assist in the boot flow and debug. These are simple registers with dual access read/write support and cause no interrupts.

30.2.5 ISH Interrupt Handling via IOAPIC (Interrupt Controller)

The legacy IOAPIC is the interrupt controller for the ISH. It collects inputs from various internal blocks and sends interrupt messages to the ISH controller. When there is a change on one of its inputs, the IOAPIC sends an interrupt message to the ISH controller.

The IOAPIC allows each interrupt input to be active high or active low and edge or level triggered.

30.3 Signal Description

Signal Name	Type	Description
GPP_B02/**ISH_I2C0_SDA**/**ISH_I3C0_SDA**/I2C2A_SDA	I/OD	ISH I²C 0 Data ISH I³C 0 Data
GPP_B03/**ISH_I2C0_SCL**/**ISH_I3C0_SCL**/I2C2A_SCL	I/OD	ISH I²C 0 Clk ISH I³C 0 Clk
GPP_H14/**ISH_UART1_RXD**/UART1A_RXD/ **ISH_I2C1_SDA**/**ISH_I3C1_SDA**	I/OD	ISH UART1 Receive Data ISH I²C 1 Data ISH I³C 1 Data
GPP_H15/**ISH_UART1_TXD**/UART1A_TXD/ **ISH_I2C1_SCL**/**ISH_I3C1_SCL**	I/OD	ISH UART1 Transmit Data ISH I²C 1 Clk ISH I³C 1 Clk
GPP_B18/**ISH_I2C2_SDA**/I2C4A_SDA	I/OD	ISH I²C 2 Data
GPP_B19/**ISH_I2C2_SCL**/I2C4A_SCL	I/OD	ISH I²C 2 Clk
GPP_D01/I2C3A_SDA/**ISH_I2C2A_SDA**	I/OD	ISH I²C 2A Data
GPP_D02/I2C3A_SCL/**ISH_I2C2A_SCL**	I/OD	ISH I²C 2A Clk
GPP_B04/BK0/SBK0/**ISH_GP0**	I/O	ISH GPIO 0
GPP_B05/BK1/SBK1/**ISH_GP1**	I/O	ISH GPIO 1
GPP_B06/BK2/SBK2/**ISH_GP2**	I/O	ISH GPIO 2
GPP_B07/BK3/SBK3/**ISH_GP3**	I/O	ISH GPIO 3
GPP_B08/BK4/SBK4/**ISH_GP4**	I/O	ISH GPIO 4
GPP_B22/TIME_SYNC0/**ISH_GP5**	I/O	ISH GPIO 5
GPP_B23/TIME_SYNC1/**ISH_GP6**	I/O	ISH GPIO 6
GPP_E05/**ISH_GP7**	I/O	ISH GPIO 7
GPP_B20/I2C5A_SDA/**ISH_GP8**	I/O	ISH GPIO 8
GPP_B21/I2C5A_SCL/**ISH_GP9**	I/O	ISH GPIO 9
GPP_E02/PROC_GP3/VRALERT#/**ISH_GP10**	I/O	ISH GPIO 10
GPP_F09/SX_EXIT_HOLDOFF#/**ISH_GP11**	I/O	ISH GPIO 11
GPP_E01/PROC_GP2/RSVD/**ISH_GP5A**	I/O	ISH GPIO 5A
GPP_F10/**ISH_GP6A**	I/O	ISH GPIO 6A
GPP_F22/THC1_DSYNC/**ISH_GP8A**	I/O	ISH GPIO 8A
GPP_F23/**ISH_GP9A**	I/O	ISH GPIO 9A

continued...

Signal Name	Type	Description
GPP_H10/UART0_RTS#/I3C1A_SDA/**ISH_GP10A**	I/O	ISH GPIO 10A
GPP_H11/UART0_CTS#/I3C1A_SCL/**ISH_GP11A**	I/O	ISH GPIO 11A
GPP_D06/**ISH_UART0_TXD**/**ISH_SPI_CLK**/SML0BCLK	O	ISH UART 0 Transmit Data ISH SPI Clock
GPP_D05/**ISH_UART0_RXD**/**ISH_SPI_CS#**/SML0BDATA	I	ISH UART 0 Receive Data ISH SPI Chip Select
GPP_D07/**ISH_UART0_RTS#**/**ISH_SPI_MISO**	O	ISH UART 0 Request To Send ISH SPI MISO
GPP_D08/**ISH_UART0_CTS#**/**ISH_SPI_MOSI**/ SML0BALERT#	I	ISH UART 0 Clear to Send ISH SPI MOSI
GPP_H07/I2C3_SCL/UART1_TXD/**ISH_UART1A_TXD**	O	ISH UART 1A Transmit Data
GPP_H06/I2C3_SDA/UART1_RXD/**ISH_UART1A_RXD**	I	ISH UART 1A Receive Data
GPP_F17/THC1_SPI2_CS#/**ISH_SPIA_CS#**/GSPI1_CS0#	O	ISH SPIA Chip Select
GPP_F11/THC1_SPI2_CLK/**ISH_SPIA_CLK**/GSPI1_CLK	O	ISH SPIA Clock
GPP_F12/THC_I2C1_SCL/I3C2_SCL/THC1_SPI2_IO0/ **ISH_SPIA_MISO**/GSPI1_MOSI/I2C5_SCL	I	ISH SPIA MISO
GPP_F13/THC_I2C1_SDA/I3C2_SDA/THC1_SPI2_IO1/ **ISH_SPIA_MOSI**/GSPI1_MISO/I2C5_SDA	O	ISH SPIA MOSI

30.4 Integrated Pull-Ups and Pull-Down

NA

30.5 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ISH_I2C0_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C0_SCL	Primary	Undriven	Undriven	Undriven
ISH_I2C1_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C1_SCL	Primary	Undriven	Undriven	Undriven
ISH_I2C2_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C2_SCL	Primary	Undriven	Undriven	Undriven
ISH_I3C0_SDA	Primary	Undriven	Undriven	Undriven
ISH_I3C0_SCL	Primary	Undriven	Undriven	Undriven
ISH_GP[11:0]	Primary	Undriven	Undriven	Undriven
ISH_GP[11:8]A ISH_GP[6:5]A	Primary	Undriven	Undriven	Undriven
ISH_UART0_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART0_RXD	Primary	Undriven	Undriven	Undriven
ISH_UART0_RTS#	Primary	Undriven	Undriven	Undriven

continued...

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ISH_UART0_CTS#	Primary	Undriven	Undriven	Undriven
ISH_UART1_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART1_RXD	Primary	Undriven	Undriven	Undriven
ISH_UART1A_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART1A_RXD	Primary	Undriven	Undriven	Undriven
ISH_SPI_CS#	Primary	Undriven	Undriven	Undriven
ISH_SPI_CLK	Primary	Undriven	Undriven	Undriven
ISH_SPI_MISO	Primary	Undriven	Undriven	Undriven
ISH_SPI_MOSI	Primary	Undriven	Undriven	Undriven
ISH_SPIA_CS#	Primary	Undriven	Undriven	Undriven
ISH_SPIA_CLK	Primary	Undriven	Undriven	Undriven
ISH_SPIA_MISO	Primary	Undriven	Undriven	Undriven
ISH_SPIA_MOSI	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

31.0 System Management Interface and SMLink

The Processor provides two SMLink interfaces, SMLink0 and SMLink1. The interfaces are intended for system management and are controlled by the Intel® CSME.

Table 94. Acronyms

Acronyms	Description
EC	Embedded Controller
BMC	Baseboard Management Controller
SPD	Serial Presence Detect
TCO	Total Cost of Ownership

31.1 Functional Description

The SMLink interfaces are controlled by the Intel® CSME.

SMLink0 is mainly used for integrated LAN. When an Intel LAN PHY is connected to SMLink0, a SMT3_EN soft strap must be set to indicate that the PHY is connected to SMLink0. The interface will be running at the frequency of up to 1 MHz depending on different factors such as board routing or bus loading when the Fast Mode is enabled using a soft strap.

SMLink1 can be used with an Embedded Controller (EC) or Baseboard Management Controller (BMC).

Both SMLink0 and SMLink1 support up to 1 MHz.

31.1.1 Integrated USB-C Usage

USBC_SML* is used to communicate with USB-C* PD Controller on the platform to configure different modes such as USB, DP, Thunderbolt etc. When used for Integrated USB-C* purposes, a soft strap must be set to indicate that integrated USB-C ports from Processor are being used.

USBC_SML uses controller mode and gets an alert signal from PMCALERT#.

Based on capabilities of different PD Controllers, re-timers needed for USB-C* connector on the platform may need to be controlled by the Processor also.

USB-C* connectors are present at one side or both side of the system,so (USBC_SML, PMCAlert) could be routed to long distance on the motherboard provided total bus capacitance specification is met.

USB-C* Re-timer control (like Firmware Load, USB-C configuration) handling depends on the number of I^2C ports available on the PD controller.

If the PD controller has two I^2C ports then Processor PMC will handle the Re-timer and PD controller .

31.2 Signal Description

Signal Name	Type	Description
GPP_C03/**SML0CLK**	I/OD	**System Management Link 0 Clock**: SMLink clock to external PHY. External Pull-up resistor required.
GPP_C04/**SML0DATA**	I/OD	**System Management Link 0 Data**: SMLink data to external PHY. External Pull-up resistor required.
GPP_C05/**SML0ALERT#**	I/OD	**System Management Link 0 Alert**: Alert for the SMLink controller to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_D06/ISH_UART0_TXD/ ISH_SPI_CLK/**SML0BCLK**	I/OD	**System Management Link 0 B Clock** External Pull-up resistor required. Note: Alternate interface from/to same SML0 controller
GPP_D05/ISH_UART0_RXD/ ISH_SPI_CS#/**SML0BDATA**	I/OD	**System Management Link 0 B Data** External Pull-up resistor required. Note: Alternate interface from/to same SML0 controller
GPP_D08/ISH_UART0_CTS#/ ISH_SPI_MOSI/**SML0BALERT#**	I/OD	**System Management Link 0 B Alert** External Pull-up resistor required. Note: Alternate interface from/to same SML0 controller
GPP_C06/**SML1CLK**	I/OD	**System Management Link 1 Clock**: SMLink clock to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_C07/**SML1DATA**	I/OD	**System Management Link 1 Data**: SMLink data to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_C08/**SML1ALERT#**/ SOCHOT#	I/OD	**System Management Link 1 Alert**: Alert for the SMLink controller to optional Embedded Controller or BMC. A soft-strap determines the native function SML1ALERT# or SOCHOT# usage. This is **NOT** the right Alert pin for USB-C* usage. External Pull-up resistor is required on this pin.
GPP_B00/**USB-C_SMLCLK**	I/OD	**System Management** bus over **Sideband 2 Core** clock External Pull-up resistor required.
GPP_B01/**USB-C_SMLDATA**	I/OD	**System Management** bus over **Sideband 2 Core** data External Pull-up resistor required.
GPP_A08/**PSE_SMLCLK**	I/OD	**Platform Security Engine System Management Link Clock**: SMLink clock to biometric sensor. External Pull-up resistor required.
GPP_A09/**PSE_SMLDATA**	I/OD	**Platform Security Engine System Management Link Data**: SMLink data to biometric sensor. External Pull-up resistor required.
GPP_A10/**PSE_SMLALERT#**	I/OD	**Platform Security Engine System Management Link Alert** : Alert for the SMLink controller to biometric sensor. External Pull-up resistor required.
INTRUDER#	I	**Intruder Detect.**

32.0 Host System Management Bus (SMBus) Controller

The Processor provides a System Management Bus (SMBus) 2.0 host controller as well as an SMBus Device Interface.

The host SMBus controller supports up to 100 kHz clock speed.

Table 95. Acronyms

Acronyms	Description
ARP	Address Resolution Protocol
CRC	Cyclic Redundancy Check
PEC	Package Error Checking
SMBus	System Management Bus

Table 96. References

Specification	Location
System Management Bus (SMBus) Specification, Version 2.0	http://www.smbus.org/specs/

32.1 Functional Description

The Processor provides an System Management Bus (SMBus) 2.0 host controller as well as an SMBus Device Interface.

- **Host Controller:** Provides a mechanism for the processor to initiate communications with SMBus peripherals (Devices).
- **Target Interface:** Allows an external host to read from or write to the Processor . Write cycles can be used to cause certain events or pass messages, and the read cycles can be used to determine the state of various status bits. The Processor 's internal host controller cannot access the Processor 's internal Device Interface.

32.1.1 Host Controller

The host SMBus controller supports up to 100 kHz clock speed and is clocked by the RTC clock.

The Processor can perform SMBus messages with either Packet Error Checking (PEC) enabled or disabled. The actual PEC calculation and checking is performed in SW. The SMBus host controller logic can automatically append the CRC byte if configured to do so.

The SMBus Address Resolution Protocol (ARP) is supported by using the existing host controller commands through software, except for the Host Notify command (which is actually a received message).

The Processor SMBus host controller checks for parity errors as a target. If an error is detected, the detected parity error bit in the PCI Status Register is set.

Host Controller Operation Overview

The SMBus host controller is used to send commands to other SMBus Target devices. Software sets up the host controller with an address, command, and, for writes, data and optional PEC; and then tells the controller to start. When the controller has finished transmitting data on writes, or receiving data on reads, it generates an SMI# or interrupt, if enabled.

The host controller supports eight command protocols of the SMBus interface (refer to the System Management Bus (SMBus) Specification, Version 2.0): Quick Command, Send Byte, Receive Byte, Write Byte/Word, Read Byte/Word, Process Call, Block Read/Write, and Block Write–Block Read Process Call.

The SMBus host controller requires that the various data and command fields be setup for the type of command to be sent. When software sets the START bit, the SMBus Host controller performs the requested transaction, and interrupts the processor (or generates an SMI#) when the transaction is completed. Once a START command has been issued, the values of the "active registers" (Host Control, Host Command, Transmit Target Address, Data 0, Data 1) should not be changed or read until the interrupt status message (INTR) has been set (indicating the completion of the command). Any register values needed for computation purposes should be saved prior to issuing of a new command, as the SMBus host controller updates all registers while completing the new command.

Target functionality, including the Host Notify protocol, is available on the SMBus pins.

Using the SMB host controller to send commands to the Processor SMB Target port is not supported.

Command Protocols

In all of the following commands, the Host Status Register (offset 00h) is used to determine the progress of the command. While the command is in operation, the HOST_BUSY bit is set. If the command completes successfully, the INTR bit will be set in the Host Status Register. If the device does not respond with an acknowledge, and the transaction times out, the DEV_ERR bit is set.

If software sets the KILL bit in the Host Control Register while the command is running, the transaction will stop and the FAILED bit will be set after the Processor forces a time - out. In addition, if KILL bit is set during the CRC cycle, both the CRCE and DEV_ERR bits will also be set.

Quick Command

When programmed for a Quick Command, the Transmit Target Address Register is sent. The PEC byte is never appended to the Quick Protocol. Software should force the PEC_EN bit to 0 when performing the Quick Command. Software must force the I2C_EN bit to 0 when running this command. Refer to Section 5.5.1 of the System Management Bus (SMBus) Specification, Version 2.0 for the format of the protocol.

Send Byte/Receive Byte

For the Send Byte command, the Transmit Target Address and Device Command Registers are sent. For the Receive Byte command, the Transmit Target Address Register is sent. The data received is stored in the DATA0 register. Software must force the I2C_EN bit to 0 when running this command.

The Receive Byte is similar to a Send Byte, the only difference is the direction of data transfer. Refer to Sections 5.5.2 and 5.5.3 of the System Management Bus *(SMBus)* Specification, Version 2.0 for the format of the protocol.

Write Byte/Word

The first byte of a Write Byte/Word access is the command code. The next 1 or 2 bytes are the data to be written. When programmed for a Write Byte/Word command, the Transmit Target Address, Device Command, and Data0 Registers are sent. In addition, the Data1 Register is sent on a Write Word command. Software must force the I2C_EN bit to 0 when running this command. Refer to Section 5.5.4 of the *System Management Bus (SMBus) Specification*, Version 2.0 for the format of the protocol.

Read Byte/Word

Reading data is slightly more complicated than writing data. First the Processor must write a command to the Target device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The target then returns 1 or 2 bytes of data. Software must force the I2C_EN bit to 0 when running this command.

When programmed for the read byte/word command, the Transmit Target Address and Device Command Registers are sent. Data is received into the DATA0 on the read byte, and the DAT0 and DATA1 registers on the read word. Refer to Section 5.5.5 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Process Call

The process call is so named because a command sends data and waits for the target to return a value dependent on that data. The protocol is simply a Write Word followed by a Read Word, but without a second command or stop condition.

When programmed for the Process Call command, the Processor transmits the Transmit Target Address, Host Command, DATA0 and DATA1 registers. Data received from the device is stored in the DATA0 and DATA1 registers.

The Process Call command with I2C_EN set and the PEC_EN bit set produces undefined results. Software must force either I2C_EN or PEC_EN to 0 when running this command. Refer to Section 5.5.6 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

NOTES

1. For process call command, the value written into bit 0 of the Transmit Target Address Register needs to be 0.
2. If the I2C_EN bit is set, the protocol sequence changes slightly, the Command Code (Bits 18:11 in the bit sequence) are not sent. As a result, the target will not acknowledge (Bit 19 in the sequence).

Block Read/Write

The Processor contains a 32 - byte buffer for read and write data which can be enabled by setting bit 1 of the Auxiliary Control register at offset 0Dh in I/O space, as opposed to a single byte of buffering. This 32 - byte buffer is filled with write data

before transmission, and filled with read data on reception. In the Processor , the interrupt is generated only after a transmission or reception of 32 bytes, or when the entire byte count has been transmitted/received.

The byte count field is transmitted but ignored by the Processor as software will end the transfer after all bytes it cares about have been sent or received.

For a Block Write, software must either force the I2C_EN bit or both the PEC_EN and AAC bits to 0 when running this command.

The block write begins with a target address and a write condition. After the command code the Processor issues a byte count describing how many more bytes will follow in the message. If a target had 20 bytes to send, the first byte would be the number 20 (14h), followed by 20 bytes of data. The byte count may not be 0. A Block Read or Write is allowed to transfer a maximum of 32 data bytes.

When programmed for a block write command, the Transmit target Address, Device Command, and Data0 (count) registers are sent. Data is then sent from the Block Data Byte register; the total data sent being the value stored in the Data0 Register.

On block read commands, the first byte received is stored in the Data0 register, and the remaining bytes are stored in the Block Data Byte register. Refer to section 5.5.7 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

NOTE

For Block Write, if the I2C_EN bit is set, the format of the command changes slightly. The Processor will still send the number of bytes (on writes) or receive the number of bytes (on reads) indicated in the DATA0 register. However, it will not send the contents of the DATA0 register as part of the message. When operating in I^2C mode (I2C_EN bit is set), the Processor will never use the 32 - byte buffer for any block commands.

I^2C* Read

This command allows the Processor to perform block reads to certain I^2C devices, such as serial E^2PROMs. The SMBus Block Read supports the 7 - bit addressing mode only.

However, this does not allow access to devices using the I^2C “Combined Format” that has data bytes after the address. Typically these data bytes correspond to an offset (address) within the serial memory chips.

NOTE

This command is supported independent of the setting of the I2C_EN bit. The I^2C Read command with the PEC_EN bit set produces undefined results. Software must force both the PEC_EN and AAC bit to 0 when running this command.

For I^2C Read command, the value written into bit 0 of the Transmit Target Address Register (SMB I/O register, offset 04h) needs to be 0.

The format that is used for the command is shown in the table below:

Table 97. I²C* Block Read

Bit	Description
1	Start
8:2	Target Address – 7 bits
9	Write
10	Acknowledge from target
18:11	Send DATA1 register
19	Acknowledge from target
20	Repeated Start
27:21	Target Address – 7 bits
28	Read
29	Acknowledge from target
37:30	Data byte 1 from target – 8 bits
38	Acknowledge
46:39	Data byte 2 from target – 8 bits
47	Acknowledge
–	Data bytes from target / Acknowledge
–	Data byte N from target – 8 bits
–	NOT Acknowledge
–	Stop

The Processor will continue reading data from the peripheral until the NAK is received.

Block Write – Block Read Process Call

The block write - block read process call is a two - part message. The call begins with a target address and a write condition. After the command code the host issues a write byte count (M) that describes how many more bytes will be written in the first part of the message. If a controller has 6 bytes to send, the byte count field will have the value 6 (0000 0110b), followed by the 6 bytes of data. The write byte count (M) cannot be 0.

The second part of the message is a block of read data beginning with a repeated start condition followed by the target address and a Read bit. The next byte is the read byte count (N), which may differ from the write byte count (M). The read byte count (N) cannot be 0.

The combined data payload must not exceed 32 bytes. The byte length restrictions of this process call are summarized as follows:

- M ≥ 1 byte
- N ≥ 1 byte
- M + N ≤ 32 bytes

The read byte count does not include the PEC byte. The PEC is computed on the total message beginning with the first target address and using the normal PEC computational rules. It is highly recommended that a PEC byte be used with the Block Write - Block Read Process Call. Software must do a read to the command register (offset 2h) to reset the 32 byte buffer pointer prior to reading the block data register.

NOTES

1. There is no STOP condition before the repeated START condition, and that a NACK signifies the end of the read transfer.
2. E32B bit in the Auxiliary Control register must be set when using this protocol.

Refer to Section 5.5.8 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Bus Arbitration

Several controllers may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The Processor continuously monitors the SMBDATA line. When the Processor is attempting to drive the bus to a 1 by letting go of the SMBDATA line, and it samples SMBDATA low, then some other controller is driving the bus and the Processor will stop transferring data.

If the Processor detects that it has lost arbitration, the condition is called a collision. The Processor will set the BUS_ERR bit in the Host Status Register, and if enabled, generates an interrupt or SMI#. The processor is responsible for restarting the transaction.

Clock Stretching

Some devices may not be able to handle their clock toggling at the rate that the Processor as an SMBus controller would like. They have the capability of stretching the low time of the clock. When the Processor attempts to release the clock (allowing the clock to go high), the clock will remain low for an extended period of time.

The Processor monitors the SMBus clock line after it releases the bus to determine whether to enable the counter for the high time of the clock. While the bus is still low, the high time counter must not be enabled. Similarly, the low period of the clock can be stretched by an SMBus controller if it is not ready to send or receive data.

Bus Timeout (Processor as SMBus controller)

If there is an error in the transaction, such that an SMBus device does not signal an acknowledge or holds the clock lower than the allowed Timeout time, the transaction will time out. The Processor will discard the cycle and set the DEV_ERR bit. The timeout minimum is 25 ms (800 RTC clocks). The Timeout counter inside the Processor will start after the first bit of data is transferred by the Processor and it is waiting for a response.

The 25 - ms Timeout counter will not count under the following conditions:

1. BYTE_DONE_STATUS bit (SMBus I/O Offset 00h, Bit 7) is set
2. The SECOND_TO_STS bit (TCO I/O Offset 06h, Bit 1) is not set (this indicates that the system has not locked up).

Interrupts/SMI#

The Processor SMBus controller uses PIRQB# as its interrupt pin. However, the system can alternatively be set up to generate SMI# instead of an interrupt, by setting the SMBUS_SMI_EN bit.

The three tables below, specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and target SMI, and Wake internal signals. The rows in the tables are additive, which means that if more than one row is true for a particular scenario then the Results for all of the activated rows will occur.

Table 98. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	SMBALERT_DIS (Target Command I/O Register, Offset 11h, Bit 2)	Result
SMBALERT# asserted low (always reported in Host Status Register, Bit 5)	X	X	X	Wake generated
	X	1	0	Target SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 99. Enables for SMBus Target Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	Event
Target Write to Wake/ SMI# Command	X	X	Wake generated when asleep. Target SMI# generated when awake (SMBUS_SMI_STS).
Target Write to SMLINK_SLAVE_SMI Command	X	X	Target SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

Table 100. Enables for the Host Notify Command

HOST_NOTIFY_INTREN (Target Control I/O Register, Offset 11h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Target Control I/O Register, Offset 11h, Bit 1)	Result
0	X	0	None
X	X	1	Wake generated
1	0	X	Interrupt generated
1	1	X	Target SMI# generated (SMBUS_SMI_STS)

SMBus CRC Generation and Checking

If the AAC bit is set in the Auxiliary Control register, the Processor automatically calculates and drives CRC at the end of the transmitted packet for write cycles, and will check the CRC for read cycles. It will not transmit the contents of the PEC register for CRC. The PEC bit must not be set in the Host Control register if this bit is set, or unspecified behavior will result.

If the read cycle results in a CRC error, the DEV_ERR bit and the CRCE bit in the Auxiliary Status register at Offset 0Ch will be set.

32.1.2 SMBus Target Interface

The Processor SMBus Target interface is accessed using the SMBus. The SMBus target logic will not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol device. The target interface allows the Processor to decode cycles, and allows an external micro controller to perform specific actions.

Key features and capabilities include:

- Supports decode of three types of messages: Byte Write, Byte Read, and Host Notify.
- Receive Target Address register: This is the address that the Processor decodes. A default value is provided so that the target interface can be used without the processor having to program this register.
- Receive Target Data register in the SMBus I/O space that includes the data written by the external micro controller.
- Registers that the external micro controller can read to get the state of the Processor .
  - — Status bits to indicate that the SMBus target logic caused an interrupt or SMI# Bit 0 of the target Status Register for the Host Notify command.
  - — Bit 16 of the SMI Status Register for all others.

NOTE

The external micro controller should not attempt to access the Processor SMBus target logic until either:

- 800 milliseconds after both: RTCRST# is high and RSMRST# is high, OR
- The PLTRST# de - asserts

If a controller leaves the clock and data bits of the SMBus interface at 1 for 50 µs or more in the middle of a cycle, the Processor target logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the target logic.

Format of Target Write Cycle

The external controller performs Byte Write commands to the Processor SMBus Target I/F. The “Command” field (bits 11:18) indicate which register is being accessed. The Data field (bits 20:27) indicate the value that should be written to that register.

The table below has the values associated with the registers.

Table 101. Target Write Registers

Register	Function
0	Command Register. Refer to the table below for valid values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–FFh	Reserved
Note: The external micro controller is responsible to make sure that it does not update the contents of the data byte registers until they have been read by the system processor. The Processor overwrites the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. The Processor will not attempt to cover this race condition (that is, unpredictable results in this case).	

Table 102. Command Types

Command Type	Description
0	Reserved
1	**WAKE/SMI#.** This command wakes the system if it is not already awake. If system is already awake, an SMI# is generated.
2	**Unconditional Powerdown.** This command sets the PWRBTNOR_STS bit, and has the same effect as the Power button Override occurring.
3	**HARD RESET WITHOUT CYCLING:** This command causes a soft reset of the system (does not include cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 2:1 set to 1, but Bit 3 set to 0.
4	**HARD RESET SYSTEM.** This command causes a hard reset of the system (including cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 3:1 set to 1.
5	**Disable the TCO Messages.** This command will disable the Processor from sending Heartbeat and Event messages. Once this command has been executed, Heartbeat and Event message reporting can only be re - enabled by assertion and then de - assertion of the RSMRST# signal.
6	**WD RELOAD:** Reload watchdog timer.
7	Reserved
8	**SMLINK_SLV_SMI.** When the Processor detects this command type while in the S0 state, it sets the SMLINK_SLV_SMI_STS bit. This command should only be used if the system is in an S0 state. If the message is received during S4 and S5 states, the Processor acknowledges it, but the SMLINK_SLV_SMI_STS bit does not get set. *Note:* It is possible that the system transitions out of the S0 state at the same time that the SMLINK_SLV_SMI command is received. In this case, the SMLINK_SLV_SMI_STS bit may get set but not serviced before the system goes to sleep. Once the system returns to S0, the SMI associated with this bit would then be generated. Software must be able to handle this scenario.
9–FFh	Reserved.

Format of Read Command

The external controller performs Byte Read commands to the Processor SMBus Target interface. The “Command” field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 103. Target Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Micro controller	
2–8	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
9	Write	External Micro controller	Always 0
10	ACK	Processor	
11–18	Command code – 8 bits	External Micro controller	Indicates which register is being accessed. Refer to the Table below for a list of implemented registers.
19	ACK	Processor	
20	Repeated Start	External Micro controller	
21–27	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
28	Read	External Micro controller	Always 1
29	ACK	Processor	
30–37	Data Byte	Processor	Value depends on register being accessed. Refer to the Table below for a list of implemented registers.
38	NOT ACK	External Micro controller	
39	Stop	External Micro controller	

Table 104. Data Values for Target Read Registers

Register	Bits	Description
0	7:0	Reserved
1	2:0	**System Power State** 000 = S0 100 = S4 101 = S5 Others = Reserved
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** *Note:* The Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the Processor will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect**. 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	Reserved
	2	Reserved
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.

continued...

Register	Bits	Description
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status.** Reflects the value of the SMBALERT# pin (when the pin is configured to SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always returns 1 if SMBALERT_DISABLE = 1.
5	0	Reserved
	1	**Battery Low Status.** 1 if the BATLOW# pin a low.
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD:** Indicates the failure core power well ramp during boot/resume. This bit will be active if the SLP_S3# pin is de - asserted and PLT_PWROK pin is not asserted.
	6	**Thermal Trip:** This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message
	7	Reserved: Default value is "X" *Note:* Software should not expect a consistent value when this bit is read through SMBUS/ SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC
E	7:0	Month of the RTC
F	7:0	Year of the RTC
10h–FFh	7:0	Reserved

- **Behavioral Notes**

  According to SMBus protocol, Read and Write messages always begin with a Start bit—Address—Write bit sequence. When the Processor detects that the address matches the value in the Receive target Address register, it will assume that the protocol is always followed and ignore the Write bit (Bit 9) and signal an Acknowledge during bit 10. In other words, if a Start—Address—Read occurs (which is invalid for SMBus Read or Write protocol), and the address matches the Processor 's Target Address, the Processor will still grab the cycle.

  Also according to SMBus protocol, a Read cycle contains a Repeated Start—Address—Read sequence beginning at Bit 20. Once again, if the Address matches the Processor 's Receive Target Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Target Read cycle.

Target Read of RTC Time Bytes

The Processor SMBus target interface allows external SMBus controller to read the internal RTC's time byte registers.

The RTC time bytes are internally latched by the Processor's hardware whenever RTC time is not changing and SMBus is idle. This ensures that the time byte delivered to the target read is always valid and it does not change when the read is still in progress on the bus. The RTC time will change whenever hardware update is in progress, or there is a software write to the RTC time bytes.

The Processor SMBus Target interface only supports Byte Read operation. The external SMBus controller will read the RTC time bytes one after another. It is the software's responsibility to check and manage the possible time rollover when subsequent time bytes are read.

For example, assuming the RTC time is 11 hours: 59 minutes: 59 seconds. When the external SMBus controller reads the hour as 11, then proceeds to read the minute, it is possible that the rollover happens between the reads and the minute is read as 0. This results in 11 hours: 0 minute instead of the correct time of 12 hours: 0 minutes. Unless it is certain that rollover will not occur, software is required to detect the possible time rollover by reading multiple times such that the read time bytes can be adjusted accordingly if needed.

Format of Host Notify Command

The Processor tracks and responds to the standard Host Notify command as specified in the *System Management Bus (SMBus) Specification,* Version 2.0. The host address for this command is fixed to 0001000b. If the Processor already has data for a previously - received host notify command which has not been serviced yet by the host software (as indicated by the HOST_NOTIFY_STS bit), then it will NACK following the host address byte of the protocol. This allows the host to communicate non - acceptance to the controller and retain the host notify address and data values for the previous cycle until host software completely services the interrupt.

NOTE

Host software must always clear the HOST_NOTIFY_STS bit after completing any necessary reads of the address and data registers.

The table below shows the Host Notify format:

Table 105. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Controller	
8:2	SMB Host Address – 7 bits	External Controller	Always 0001_000
9	Write	External Controller	Always 0
10	ACK (or NACK)	Processor	Processor NACKs if HOST_NOTIFY_STS is 1
17:11	Device Address – 7 bits	External Controller	Indicates the address of the controller ; loaded into the Notify Device Address Register
18	Unused – Always 0	External Controller	7 - bit - only address; this bit is inserted to complete the byte

continued...

Bit	Description	Driven By	Comment
19	ACK	Processor	
27:20	Data Byte Low – 8 bits	External Controller	Loaded into the Notify Data Low Byte Register
28	ACK	Processor	
36:29	Data Byte High – 8 bits	External Controller	Loaded into the Notify Data High Byte Register
37	ACK	Processor	
38	Stop	External Controller	

Format of Read Command

The external controller performs Byte Read commands to the Processor SMBus Target interface. The "Command" field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 106. Target Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Micro controller	
2–8	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
9	Write	External Micro controller	Always 0
10	ACK	Processor	
11–18	Command code – 8 bits	External Micro controller	Indicates which register is being accessed. Refer to the Tale below for a list of implemented registers.
19	ACK	Processor	
20	Repeated Start	External Micro controller	
21–27	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
28	Read	External Micro controller	Always 1
29	ACK	Processor	
30–37	Data Byte	Processor	Value depends on register being accessed. Refer to the Table below for a list of implemented registers.
38	NOT ACK	External Micro controller	
39	Stop	External Micro controller	

Table 107. Data Values for Target Read Registers

Register	Bits	Description
0	7:0	Reserved for capabilities indication. Should always return 00h. Future chips may return another value to indicate different capabilities.
1	2:0	**System Power State** 000 = S0 100 = S4 101 = S5 Others = Reserved

continued...

Register	Bits	Description
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** *Note:* The Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the Processor will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect.** 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	**Temperature Event.** 1 = Temperature Event occurred. This bit will be set if the Processor 's THRM# input signal is active. Else this bit will read "0."
	2	**DOA Processor Status**. This bit will be 1 to indicate that the processor is dead
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status:** Reflects the value of the GPIO11/SMBALERT# pin (when the pin is configured as SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always return 1 if SMBALERT_DISABLE = 1. (high = 1, low = 0).
5	0	**FWH bad bit:** This bit will be 1 to indicate that the FWH read returned FFh, which indicates that it is probably blank.
	1	**Battery Low Status:** 1 if the BATLOW# pin is a 0.
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD:** Indicates the failure core power well ramp during boot/resume. This bit will be active if the PLT_PWROK pin is not asserted.
	6	**Thermal Trip:** This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message.
	7	Reserved: Default value is "X" *Note:* Software should not expect a consistent value when this bit is read through SMBUS/ SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC

continued...

Register	Bits	Description
E	7:0	Month of the RTC
F	7:0	Year of the RTC
10h–FFh	7:0	Reserved

Table 108. Enables for SMBus Target Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	Event
Target Write to Wake/SMI# Command	X	X	Wake generated when asleep. Target SMI# generated when awake (SMBUS_SMI_STS)
Target Write to SMLINK_SLAVE_SMI Command	X	X	Target SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

32.2 SMBus Power Gating

SMBus shares the Power Gating Domain with Primary-to-Sideband Bridge (P2SB). A single FET controls the single Power Gating Domain; but SMBus and P2SB each has its own dedicated Power Gating Control Block. The FET is only turned off when all these interfaces are ready to PG entry or already in the PG state.

32.3 Signal Description

Signal Name	Type	Description	Availability
GPP_C00/**SMBCLK**	I/OD	**SMBus Clock:** External Pull-up resistor is required.	All
GPP_C01/ **SMBDATA**	I/OD	**SMBus Data:** External Pull-up resistor is required.	All
GPP_C02/ **SMBALERT#**	I/OD	**SMBus Alert:** This signal is used to wake the system or generate SMI#. External Pull-up resistor is required.	All

32.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SMBALERT#	Pull-down	20 kohm ± 30%	The internal pull-down resistor is disable after RSMRST# de-asserted.

32.5 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
SMBDATA	Primary	Undriven	Undriven	Undriven
SMBCLK	Primary	Undriven	Undriven	Undriven
SMBALERT#	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

33.0 Serial Peripheral Interface (SPI)

The processor provides Serial Peripheral Interfaces (SPI) to connect up to two flash devices. The SPI0 interface consists of three Chip Select signals. SPI0 interface can allow two flash memory devices (SPI0_CS0# and SPI0_CS1#) and one TPM device (SPI0_CS2#) to be connected to the processor. The SPI0 interface supports 1.8 V.

Table 109. Acronyms

Acronyms	Description
CLK	Clock
CS	Chip Select
FCBA	Flash Component Base Address
FLA	Flash Linear Address
FMBA	Flash Controller Base Address
FPSBA	Flash Processor Strap Base Address
FRBA	Flash Region Base Address
MDTBA	MIP Descriptor Table Base Address
MISO	Terminology to indicate signal direction: input to the host, output from the device
MOSI	Terminology to indicate signal direction: output from the host, input to the device
TPM	Trusted Platform Module

33.1 Functional Description

33.1.1 SPI0 Support for TPM

The processorSPI0 flash controller supports a discrete TPM on the platform via its dedicated SPI0_CS2# signal. The platform must have no more than 1 TPM.

The SPI0 controller default reset frequency is 20 MHz, but a valid soft strap setting can override this for required operating frequency. The SPI0 TPM device must support a 20 MHz clock and should be able to operate within the 15-20 MHz range. Support for frequencies above 20 MHz is optional. The SPI0 controller supports a maximum operating frequency of 48 MHz for dTPM.

TPM requires the support for the interrupt routing. However, the TPM’s interrupt pin is routed to the processor interrupt configurable GPIO pin. Thus, TPM interrupt is completely independent from the SPI0 controller.

33.1.2 SPI0 for Flash

The Serial Peripheral Interface (SPI0) supports two SPI flash devices via two chip select (SPI0_CS0# and SPI0_CS1#). The maximum size of flash supported is determined by the SFDP-discovered addressing capability of each device. Each component can be up to 16 MB (32 MB total addressable) using 3-byte addressing. Each component can be up to 64 MB (128 MB total addressable) using 4-byte addressing. Another chip select (SPI0_CS2#) is also available and only used for TPM on SPI support. The SPI interface supports 1.8 V only. The SPI0 controller supports a maximum operating frequency of 50 MHz.

A SPI0 flash device supporting SFDP (Serial Flash Discovery Parameter) is required for all design. A SPI0 flash device on SPI0_CS0# with a valid descriptor must be attached directly to the processor.

The processor supports fast read which consist of:

1. Dual Output Fast Read (Single Input Dual Output)
2. Dual I/O Fast Read (Dual Input Dual Output)
3. Quad Output Fast Read (Single Input Quad Output)
4. Quad I/O Fast Read (Quad Input Quad Output)

The processor SPI0 has a third chip select SPI0_CS2# for TPM support over SPI. The TPM on SPI0 will use SPI0_CLK, SPI0_MISO, SPI0_MOSI and SPI0_CS2# SPI signals.

SPI0 Supported Features

- **Descriptor Mode**

  Descriptor Mode is required for all SKUs of the processor. Non-Descriptor Mode is not supported.

- **SPI0 Flash Regions**

In Descriptor Mode the Flash is divided into five separate regions.

Table 110. SPI0 Flash Regions

Region	Content
0	Flash Descriptor
1	BIOS
2	Intel ®CSME
3	GbE - Location for Integrated LAN firmware and MAC address
4	PDR - Platform Data Region
8	EC - Embedded Controller
10	Intel® Silicon Security Engine

Only four controllers can access the regions: Host processor running BIOS code, Integrated Gigabit Ethernet and Host processor running Gigabit Ethernet Software, Intel Converged Security and Management Engine, and the EC.

The Flash Descriptor and Intel® CSME region are the only required regions. The Flash Descriptor has to be in region 0 and region 0 must be located in the first sector of Device 0 (Offset 0). All other regions can be organized in any order.

Regions can extend across multiple components, but must be contiguous.

Flash Region Sizes

SPI0 flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4 KB or larger block. GbE requires two 4 KB or larger blocks. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the Intel® CSME and BIOS regions. The Intel® CSME region contains firmware to support Intel Active Management Technology and other Intel® CSME capabilities.

Table 111. Region Size Versus Erase Granularity of Flash Components

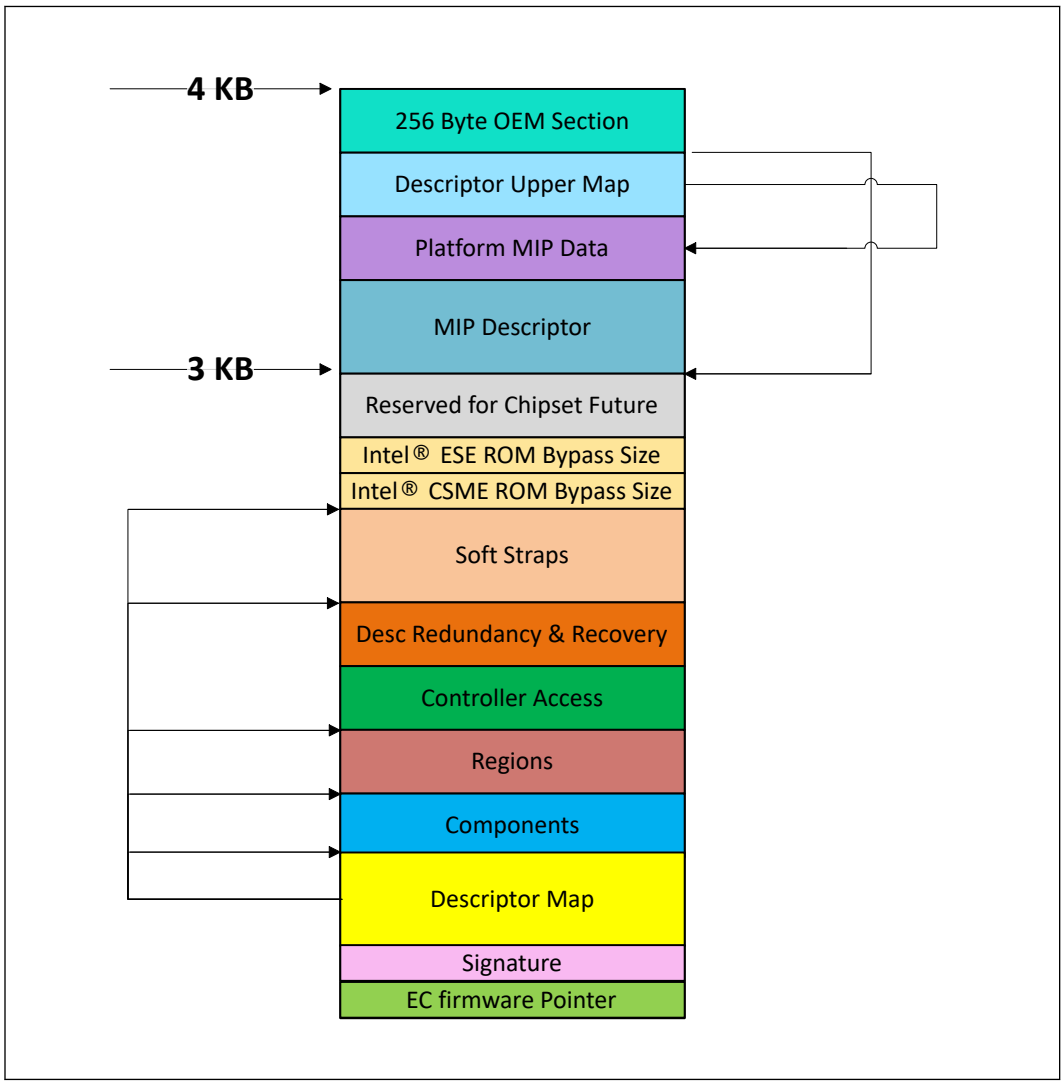
Region	Size with 4 KB Blocks	Size with 8 KB Blocks	Size with 64 KB Blocks
Descriptor	4 KB	8 KB	64 KB
GbE	8 KB	16 KB	128 KB
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSME	Varies by Platform	Varies by Platform	Varies by Platform
EC	Varies by Platform	Varies by Platform	Varies by Platform
PDR	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSME Data	Varies by Platform	Varies by Platform	Varies by Platform

Flash Descriptor

The bottom sector of the flash component 0 contains the Flash Descriptor. The maximum size of the Flash Descriptor is 4 KB. If the block/sector size of the SPI0 flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. It requires its own discrete erase block, so it may need greater than 4 KB of flash space depending on the flash architecture that is on the target system. Two additional redundant back-ups of the Flash Descriptor have been added for data resilience.The information stored in the Flash Descriptor can only be written during the manufacturing process as its read/write permissions must be set to read only when the computer leaves the manufacturing floor.

The Flash Descriptor is made up of fifteen sections as shown in the figure below:

Figure 16. Flash Descriptor Regions



- EC Firmware Pointer is located in the first 16 bytes of the Descriptor and contains the address location for EC flash region. The format for the EC Firmware Pointer address is dependent on EC vendors/OEM implementation of this field.
- The Flash signature at the bottom of the flash (offset 10h) must be 0FF0A55Ah in order to be in Descriptor mode.
- The Descriptor map has pointers to the lower five descriptor sections as well as the size of each.
- The Component section has information about the SPI flash part(s) the system. It includes the number of components, density of each component, read, write and erase frequencies and invalid instructions.
- The Region section defines the base and the limit of the BIOS, IFWI, GbE, Platform Data Region (PDR- Optional), Embedded Controller (EC- Optional) regions as well as their size.
- The processor soft strap sections contain configurable parameters.

- The Reserved region is for future processor usage.
- The Descriptor Upper Map determines the length and base address of the Intel ® CSME VSCC Table.
- The Intel ® CSME VSCC Table holds the JEDEC ID and the ME VSCC information for all the SPI Flash part(s) supported by the NVM image. BIOS and GbE write and erase capabilities depend on VSCC0 and VSCC1 registers in SPIBAR memory space.
- OEM Section is 256 bytes reserved at the top of the Flash Descriptor for use by OEM.

Descriptor Controller Region

The Controller region defines read and write access setting for each region of the SPI0 device. The Controller region recognizes four Controllers: BIOS, Gigabit Ethernet, Intel® CSME, and EC. Each Controller is only allowed to do direct reads of its primary regions.

Table 112. Region Access Control Table

Controller Read/Write Access				
Region	**Processor and BIOS**	**Intel® CSME**	**GbE Controller**	**EC**
Descriptor (0)	Read Only	Read Only	Not Accessible	Not Accessible
BIOS (1)	processor / BIOS can always read from and write to BIOS region prior to EOP	Not Accessible	Not Accessible	Not Accessible
Intel® CSME (2)	Read/Write (BIOS Only)	Intel® CSME can always read from and write to firmware region	Not Accessible	Not Accessible
Gigabit Ethernet (3)	Not Accessible	Read/Write	GbE software can always read from and write to GbE region	Not Accessible
PDR (4)	Not Accessible	Not Accessible	Not Accessible	Not Accessible
EC (8)	Read/Write	Not Accessible	Not Accessible	EC can always read from and write to EC region.
Intel® CSME Data (15)	Not Accessible	Read/Write	Not Accessible	Not Accessible

Notes:
- The Region Access values listed above represent post manufacturing configuration only.
- Descriptor and PDR region is not a Controller, so they will not have Controller R/W access.
- Descriptor should NOT have write access by any Controller in production systems.
- PDR region should only have read and/or write access by processor/Host. GbE and Intel® CSME should NOT have access to PDR region.

Table 113. Flash Descriptor Processor Complex Soft Strap

Region Name	Starting Address
Signature	10h
Component FCBA	30h
Regions FRBA	40h
Controllers FMBA	80h

continued...

Region Name	Starting Address
Desc Redundancy & Recovery	320h
MDTBA	C00h
Processor Straps	CECh D8Ch
Intel® CSME Straps	D9Ch

Flash Access

There are two types of accesses: Direct Access and Program Register Accesses.

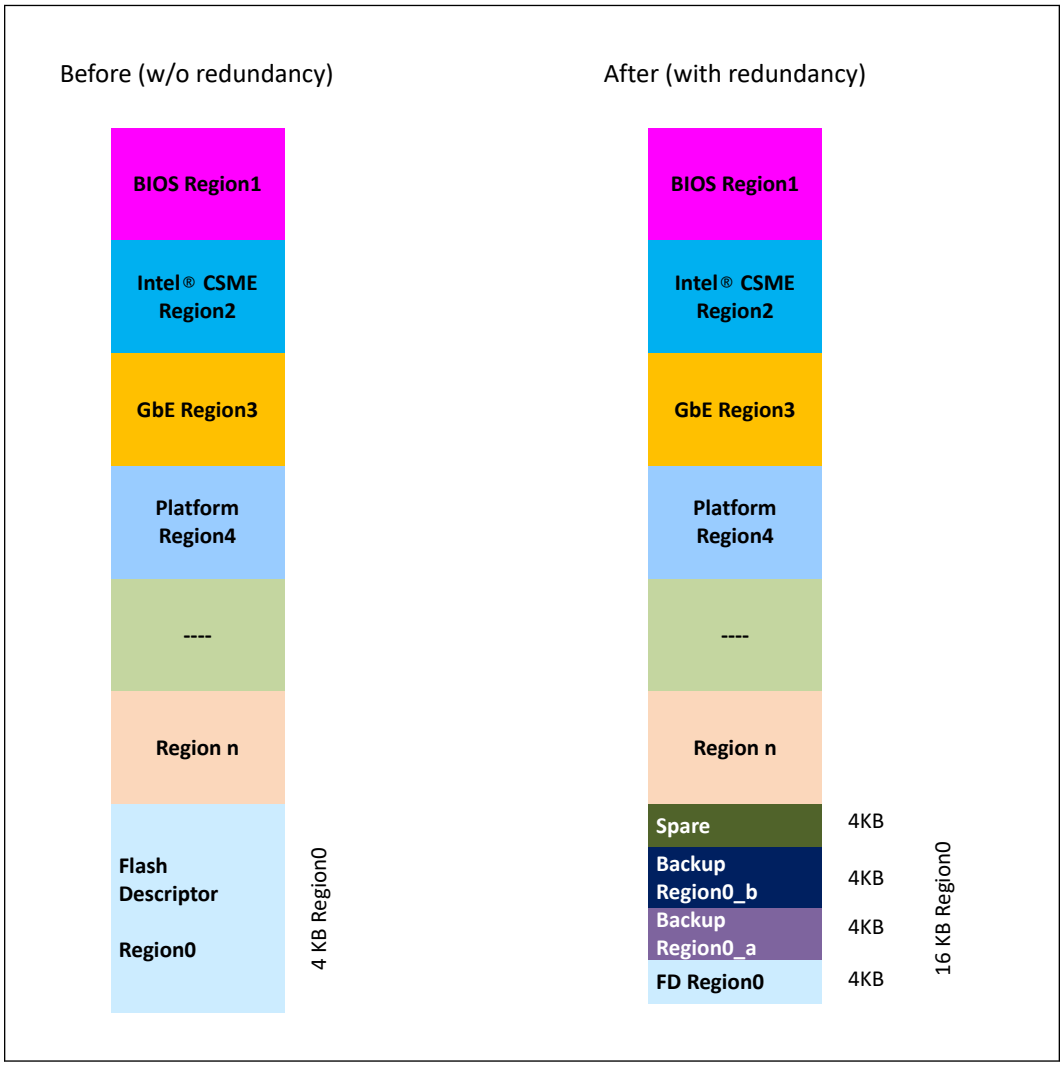
- **Direct Access**
  - Controllers are allowed to do direct read only of their primary region
    - Gigabit Ethernet region can only be directly accessed by the Gigabit Ethernet controller. Gigabit Ethernet software must use Program Registers to access the Gigabit Ethernet region.
  - Controller's Host or Management Engine virtual read address is converted into the SPI0 Flash Linear Address (FLA) using the Flash Descriptor Region Base/ Limit registers

    Direct Access Security
  - Requester ID of the device must match that of the primary Requester ID in the Controller Section
  - Calculated Flash Linear Address must fall between primary region base/limit
  - Direct Write not allowed
  - Direct Read Cache contents are reset to 0's on a read from a different Controller
- **Program Register Access**
  - Program Register Accesses are not allowed to cross a 4 KB boundary and cannot issue a command that might extend across two components
  - Software programs the FLA corresponding to the region desired
    - Software must read the devices Primary Region Base/Limit address to create a FLA.
  - *Register Access Security*
    Only primary region Controllers can access the registers

Flash Descriptor Redundancy and Recovery

In order to provide descriptor redundancy and recovery, SPI flash controller uses two 4 KB spaces or regions as the backup descriptor regions. Each backup descriptor region size is 4 KB.

Figure 17. Flash Descriptor Redundancy



In the main and backup descriptor regions, the following fields are defined for the descriptor integrity check and recovery. Before SPI controller reads the descriptor, it

- Reads Main Descriptor Region and calculates SHA-256 hash.
- Reads Active Backup Descriptor Region and calculates hash.
- Compares each hash result with the hash in that region.
- Takes action based on result and policy byte (in Main Descriptor).

RPMC Configuration

Intel Replay Protection Monotonic Counter (RPMC) is a capability providing Anti-Replay Protection using Monotonic Counters inside SPI Flash. Intel RPMC is a critical security feature designed to protect the SPI part of Intel platforms from unauthorized write operations. This innovative technology acts as a robust defense mechanism, ensuring that only authorized write operations are permitted, thus preventing any unauthorized access to the SPI.

RPMC protection relies on:

- Special RPMC HW and logic inside the SPI Flash.
- Intel CSME FW support that utilizes RPMC capabilities within Flash.

At the core of RPMC's functionality lies the concept of the session key.

The session key is a cryptographic key derived from several factors residing on the processor . These factors are carefully selected and stored upon provisioning RPMC to the SPI part. The session key serves as a means of authenticating each incoming write message to the SPI. When an authorized operation is initiated, the session key is used to verify the legitimacy of the request. If the session key does not match the expected value, the SPI part will reject the request, effectively blocking malicious or unauthorized write operations.

Furthermore, the session key also extends its protective shield to cover a specific set of sensitive read messages. This holistic approach ensures that not only write operations but also read operations involving sensitive data are monitored and authenticated, enhancing the overall security of the system.

Two features of RPMC can be enabled:

- RPMC will be enabled on platforms with RPMC SPI. During Intel End of Manufacturing the processor will be bound with RPMC SPI
- When SPI is replaced, re-binding between the new RPMC SPI and the processor will happen automatically on first boot.

Monotonic Counters

Monotonic counters are counters on the SPI Flash maintained by Intel CSME FW. SPI Flash has a set of four 32-bit monotonic counters, where Intel CSME FW uses two of these counters. Intel CSME FW ensures FW write operations will not exceed SPI RPMC monotonic counter increment rate specified by RPMC HW during platform lifetime supported by Intel. Reading and incrementing the counters in the Flash is done using authenticated commands with a key known to both: SPI Flash and Intel® CSME FW

Binding at End of Manufacturing (EOM)

RPMC Binding pairs between SPI Flash and the processor by provisioning the Binding key produced by the processor into SPI Flash. This pairing is done as part of the EOM flow which usually takes place at the manufacturing line.

In conclusion, Intel RPMC, with its Replay Monotonic Counter and session key mechanism, stands as a powerful safeguard against unauthorized write operations and unauthorized access to sensitive data in the SPI part. This robust security feature, derived from the session key, adds an additional layer of protection to Intel platforms, making them more resilient against potential threats and ensuring the integrity and confidentiality of the data stored in the SPI.

33.2 Signal Description

Signal Name	Type	Description
SPI0_CLK	O	**SPI0 Clock**: SPI clock signal for the common flash/TPM interface.
SPI0_CS0#	O	**SPI0 Chip Select 0**: Used to select the primary SPI0 Flash device. *Note:* This signal cannot be used for any other type of device than SPI Flash.
		continued...

Signal Name	Type	Description
SPI0_CS1#	O	**SPI0 Chip Select 1**: Used to select an optional secondary SPI0 Flash device. *Note:* This signal cannot be used for any other type of device than SPI Flash.
SPI0_CS2#	O	**SPI0 Chip Select 2**: Used to select the TPM device if it is connected to the SPI0 interface. It cannot be used for any other type of device.
SPI0_MOSI	I/O	**SPI0 Host OUT Device IN**: Defaults as a data output pin for the processor in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI0_IO0) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_MISO	I/O	**SPI0 Host IN Device OUT**: Defaults as a data input pin for the processor in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI0_IO1) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_IO2	I/O	**SPI0 Data I/O**: A bidirectional signal used to support Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes. This signal is not used in Dual Output Fast Read mode.
SPI0_IO3	I/O	**SPI0 Data I/O**: A bidirectional signal used to support Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes. This signal is not used in Dual Output Fast Read mode.

33.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SPI0_CLK	Pull-down	20 kohm ± 30%	
SPI0_MOSI	Pull-up	20 kohm ± 30%	Note
SPI0_MISO	Pull-up	20 kohm ± 30%	Note
SPI0_CS[2:0]#	Pull-down	20 kohm ± 30%	
SPI0_IO[2:3]	Pull-up	20 kohm ± 30%	

NOTE

Above resistor type is dynamic state controlled by the SPI controller. The internal Pull-up is disabled when RSMRST# is asserted (during reset) and only enabled after RSMRST# de-assertion.

33.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
SPI0_CLK	Primary	Internal Pull-down	Driven Low	Driven Low
SPI0_MOSI	Primary	Hi-Z	Internal Pull-up , then Driven Low	Driven Low
SPI0_MISO	Primary	Hi-Z	Internal Pull-up	Internal Pull-up
SPI0_CS[2:0]#	Primary	Internal Pull-down	Driven High	Driven High
SPI0_IO[3:2]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up
Note: 1. During reset refers to when RSMRST# is asserted.				

34.0 Enhanced Serial Peripheral Interface (eSPI)

The processor provides the Enhanced Serial Peripheral Interface (eSPI) to support connection of an EC (typically used in mobile platform) or an SIO (typically used in desktop platform) to the platform. Below are the key features of the interface:

- 1.8 V support only
- Support for Host Attached Flash (MAF) and Device Attached Flash (SAF).
- Support for up to 50 MHz (configured by soft straps)
- Up to quad mode support
- Support for PECI over eSPI
- Support for Multiple OOB Controller (dedicated OOB channel for different OOB Controllers in the Processor such as PMC and CSME)
- Transmitting RTC time/date to the device upon request
- In-band messages for communication between the Processor and device to eliminate side-band signals.
- Real time SPI flash sharing, allowing real time operational access by the processor and device.

Table 114. Acronyms

Acronyms	Description
EC	Embedded Controller
MAFCC	Host Attached Flash Channel Controller (MAFCC)
SAFCC	Device Attached Flash Channel Controller (SAFCC)
OOB	Out-of-Band
TAR	Turn-around cycle

Table 115. References

Specification	Document Number/Location
Enhanced Serial Peripheral Interface (eSPI) Specifications	https://downloadcenter.intel.com/download/27055/eSPI

34.1 Functional Description

34.1.1 Channels and Supported Transactions

NOTE

Flash channel access is POR exclusively on ESPI_CS0#.

NOTE

ESPI_CS0# supports up to four Generic I/O ranges, while the other chip selects can support only one Generic I/O range each.

An eSPI channel provides a means to allow multiple independent flows of traffic to share the same physical bus. Refer to the eSPI specification for more detail.

Each of the channels has its dedicated resources such as queue and flow control. There is no ordering requirement between traffic from different channels.

The number of types of channels supported by a particular eSPI device is discovered through the GET_CONFIGURATION command issued by the processor to the eSPI device during initialization.

Table below summarizes the eSPI channels and supported transactions.

Table 116. eSPI Channels and Supported Transactions

CH #	Channel	Posted Cycles Supported	Non-Posted Cycles Supported
0	Peripheral	Memory Write, Completions	Memory Read, I/O Read/Write
1	Virtual Wire	Virtual Wire GET/PUT	N/A
2	Out-of-Band Message	SMBus Packet GET/PUT	N/A
3	Flash Access	N/A	Flash Read, Write, Erase
N/A	General	Register Accesses	N/A

Peripheral Channel (Channel 0) Overview

The Peripheral channel performs the following functions:

- **Target for PCI Device**: The eSPI controller duplicates the legacy LPC PCI Configuration space registers. These registers are mostly accessed via the BIOS, though some are accessed via the OS as well.
- **Tunnel all Host to eSPI device (EC/SIO) Debug Device Accesses**: These are the accesses that used to go over the LPC bus. These include various programmable and fixed I/O ranges as well as programmable Memory ranges. The programmable ranges and their enables reside in the PCI Configuration space.
- **Tunnel all Accesses from the eSPI device to the Host**: These include Memory Reads and Writes.

Virtual Wire Channel (Channel 1) Overview

The Virtual Wire channel uses a standard message format to communicate several types of signals between the components on the platform.

- **Sideband and GPIO Pins**: System events and other dedicated signals between the processor and eSPI device. These signals are tunneled between the 2 components over eSPI.
- **Serial IRQ Interrupts**: Interrupts are tunneled from the eSPI device to the processor. Both edge and triggered interrupts are supported.
- **eSPI Virtual Wires (VW)**

  Table below summarizes the virtual wires in eSPI mode.

Table 117. eSPI Virtual Wires (VW)

Virtual Wire	processor Pin Direction	Reset Control	Pin Retained in processor (For Use by Other Components)
SUS_STAT#	Output	ESPI_RESET#	No
PRIM_PWRDN_ACK	Output	ESPI_RESET#	No
SUSWARN#	Output	ESPI_RESET#	No
SUS_ACK	Input	ESPI_RESET#	No
PLTRST#	Output	ESPI_RESET#	Yes
PME# (eSPI Peripheral PME)	Input	ESPI_RESET#	N/A
WAKE#	Input	ESPI_RESET#	No
SMI#	Input	PLTRST#	N/A
SCI#	Input	PLTRST#	N/A
RCIN#	Input	PLTRST#	No
SLP_A#	Output	ESPI_RESET#	Yes
SLP_S3#/SLP_S4#/SLP_S5#/ SLP_LAN#	Output	RSMRST#	Yes
DEVICE_BOOT_LOAD_DONE	Input	ESPI_RESET#	N/A
DEVICE_BOOT_LOAD_STATUS	Input	ESPI_RESET#	N/A
HOST_RST_WARN	Output	PLTRST#	N/A
HOST_RST_ACK	Input	PLTRST#	N/A
OOB_RST_WARN	Output	ESPI_RESET#	N/A
OOB_RST_ACK	Input	ESPI_RESET#	N/A
HOST_C10	Output	PLTRST#	N/A
ERROR_NONFATAL	Input	ESPI_RESET#	N/A
ERROR_FATAL	Input	ESPI_RESET#	N/A
DNX_WARN	Output	PLTRST#	N/A
DNX_ACK	Input	ESPI_RESET#	N/A

- **Interrupt Events**

  eSPI supports both level and edge-triggered interrupts. Refer to the eSPI Specification for details on the theory of operation for interrupts over eSPI.

  The eSPI controller will issue a message to the interrupt controller when it receives an IRQ group in its VW packet, indicating a state change for that IRQ line number.

  The eSPI device can send multiple VW IRQ index groups in a single eSPI packet, up to the Operating Maximum VW Count programmed in its Virtual Wire Capabilities and Configuration Channel.

  The eSPI controller acts only as a transport for all interrupt events generated from the device. It does not maintain interrupt state, polarity or enable for any of the interrupt events.

Out-of-Band Channel (Channel 2) Overview

The Out-of-Band channel performs the following functions:

- **Tunnel MCTP Packets between the Intel® CSME and eSPI Device**: The Intel® CSME communicates MCTP messages to/from the device by embedding those packets over the eSPI protocol. This eliminates the SMBus connection between the processor and the device which was used to communicate the MCTP messages. The eSPI controller simply acts as a message transport and forwards the packets between the Intel® CSME and eSPI device.
- **Tunnel Processor Temperature Data to the eSPI device**: The eSPI controller stores the processor temperature data internally and sends it to the device using a posted OOB message when a request is made to a specific destination address.
- **Tunnel Processor RTC Time and Date Bytes to the eSPI device**: the eSPI controller captures this data internally at periodic intervals from the processor RTC controller and sends it to the device using a posted OOB message when a request is made to a specific destination address.
- **Processor Temperature Data Over eSPI OOB Channel**

  eSPI controller supports the transmitting of processor thermal data to the eSPI device. The thermal data consists of 1 byte of processor temperature data that is transmitted periodically (~1 ms) from the thermal sensor unit.

  The packet formats for the temperature request from the eSPI device and the processor response back are shown in the two figures below.

Figure 18. eSPI Device Request to Processor for Processor Temperature

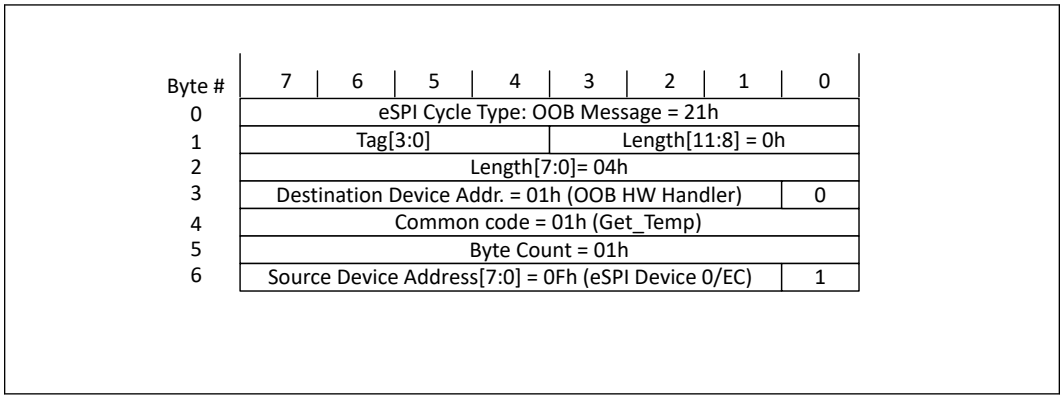


Figure 19. Processor Response to eSPI Device with Processor Temperature

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 05h							
3	Destination Device Addr. = 0Eh (eSPI Device 0/EC)							0
4	Common code = 01h (Get_Temp)							
5	Byte Count = 02h							
6	Source Device Address [7:0] = 01h (OOB HW Handler)							1
7	Temperature Data [7:0]							

- **Processor RTC Time/Date to EC Over eSPI OOB Channel**

  The processor eSPI controller supports the transmitting of processor RTC time/date to the eSPI device. This allows the eSPI device to synchronize with the Processor RTC system time. Moreover, using the OOB message channel allows reading of the internal time when the system is in Sx states.

  The RTC time consists of 7 bytes: seconds, minutes, hours, day of week, day of month, month and year. The controller provides all the time/date bytes together in a single OOB message packet. This avoids the boundary condition of possible roll over on the RTC time bytes if each of the hours, minutes, and seconds bytes is read separately.

  The packet formats for the RTC time/date request from the eSPI device and the processor response back to the device are shown in the two figures below.

Figure 20. eSPI Device Request to Processor for Processor RTC Time

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 04h							
3	Dest Device Addr. [7:1] = 01h (OOB HW Handler)							0
4	Common code = 02h (Get_RTC_Time)							
5	Byte Count = 01h							
6	Source Device Address [7:0] = 0Fh (eSPI Device 0/EC)							1

Figure 21. Processor Response to eSPI device with RTC Time

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 0Ch							
3	Dest Device Addr. [7:0] = 0Eh (eSPI Device 0/EC)							0
4	Common code = 02h (Get_RTC_Time)							
5	Byte Count = 09h							
6	Source Device Address [7:1] = 01h (OOB HW Handler)							1
7	Reserved					DM	HF	DS
8	RTC Time: Seconds							
9	RTC Time: Minutes							
10	RTC Time: Hours							
11	RTC Time: Day of Week							
12	RTC Time: Day of Month							
13	RTC Time: Month							
14	RTC Time: Year							

NOTES

1. **DS**: Daylight Savings. A 1 indicates that Daylight Saving has been comprehended in the RTC time bytes. A 0 indicates that the RTC time bytes do not comprehend the Daylight Savings.
2. **HF**: Hour Format. A 1 indicates that the Hours byte is in the 24-hr format. A 0 indicates that the Hours byte is in the 12-hr format. In 12-hr format, the seventh bit represents AM when it is a 0 and PM when it is a 1.
3. **DM**: Data Mode. A 1 indicates that the time byte are specified in binary. A 0 indicates that the time bytes are in the Binary Coded Decimal (BCD) format.

Flash Access Channel (Channel 3) Overview

The Flash Access channel supports the Host Attached Flash (MAF) configuration, where the flash device is directly attached to the processor. This configuration allows the eSPI device to access the flash device attached to the processor through a set of flash access commands. These commands are routed to the flash controller and the return data is sent back to the eSPI device.

The Host Attached Flash Channel controller (MAFCC) tunnels flash accesses from eSPI device to the flash controller. The MAFCC simply provides Flash Cycle Type, Address, Length, Payload (for writes) to the flash controller. The flash controller is responsible for all the low level flash operations to perform the requested command and provides a return data/status back to the MAFCC, which then tunnels it back to the eSPI device in a separate completion packet.

- **Host Attached Flash Channel Controller (MAFCC) Flash Operations and Addressing**

  The EC is allocated a dedicated region within the eSPI Host-Attached flash device. The EC has default read, write, and erase access to this region.

The EC can also access any other flash region as permitted by the Flash Descriptor settings. As such, the EC uses linear addresses, valid up to the maximum supported flash size, to access the flash.

The MAFCC supports flash read, write, and erase operations only.

- **Device Attached Flash Channel Controller (SAFCC) Flash Operation and Addressing**

  The processor is allocated dedicated regions (for each of the supported Controllers) within the eSPI SAFCC. The processor has read, write, and erase access to these regions, as well as any other regions that maybe permitted by the region protections set in the Flash Descriptor.

  The Device will optionally perform additional checking on the processor provided address. In case of an error due to incorrect address or any other issues it will synthesize an unsuccessful completion back to the eSPI Host.

  The SAFCC supports Flash Read, Write and Erase operations. It also supports Read SFDP and Read JEDEC ID commands as specified in the eSPI Specification for Server platforms.

34.2 Signal Description

Signal Name	Type	Description	Availability
GPP_A00/**ESPI_IO0**	I/O	**eSPI Data Signal 0:** Bi-directional pin used to transfer data between the Processor and eSPI device.	All Processor Series
GPP_A01/**ESPI_IO1**	I/O	**eSPI Data Signal 1:** Bi-directional pin used to transfer data between the Processor and eSPI device	All Processor Series
GPP_A02/**ESPI_IO2**/ PRIMPWRDNACK	I/O	**eSPI Data Signal 2:** Bi-directional pin used to transfer data between the Processor and eSPI device	All Processor Series
GPP_A03/**ESPI_IO3**/ PRIMACK#	I/O	**eSPI Data Signal 3:** Bi-directional pin used to transfer data between the Processor and eSPI device	All Processor Series
GPP_A04/**ESPI_CS0#**	O	**eSPI Chip Select 0:** Driving CS# signal low to select eSPI device for the transaction.	All Processor Series
GPP_A05/**ESPI_CLK**	O	**eSPI Clock:** eSPI clock output from the Processor to device.	All Processor Series
GPP_A06/**ESPI_RESET#**	O	**eSPI Reset:** Reset signal from the Processor to eSPI device.	All Processor Series

35.0 Intel® Serial IO Generic SPI (GSPI) Controllers

The Processor implements three generic SPI interfaces to support devices that uses serial protocol for transferring data.

Each interface consists of a clock (CLK), one chip selects (CS) and two data lines (MOSI and MISO).

The GSPI interfaces support the following features:

- Support bit rates up to 20 Mbits/s
- Support data size from 4 to 32 bits in length and FIFO depths of 64 entries
- Support DMA with 128-byte FIFO per channel (up to 64-byte burst)
- Full duplex synchronous serial interface
- Support the Motorola's* SPI protocol
- Operate in Host mode only

NOTE

Device mode is not supported.

Table 118. Acronyms

Acronyms	Description
GSPI	Generic Serial Peripheral Interface
LTR	Latency Tolerance Reporting

35.1 Functional Description

35.1.1 Controller Overview

The generic SPI controllers can only be set to operate as a Host.

The processor or DMA accesses data through the GSPI port's transmit and receive FIFOs.

A processor access takes the form of programmed I/O, transferring one FIFO entry per access. Processor accesses must always be 32 bits wide. Processor writes to the FIFOs are 32 bits wide, but the Processor will ignore all bits beyond the programmed FIFO data size. Processor reads to the FIFOs are also 32 bits wide, but the receive data written into the Receive FIFO is stored with '0' in the most significant bits (MSB) down to the programmed data size.

The FIFOs can also be accessed by DMA, which must be in multiples of 1, 2, or 4 bytes, depending upon the EDSS value, and must also transfer one FIFO entry per access.

For writes, the Enhanced SPI takes the data from the transmit FIFO, serializes it, and sends it over the serial wire to the external peripheral. Receive data from the external peripheral on the serial wire is converted to parallel words and stored in the receive FIFO.

A programmable FIFO trigger threshold, when exceeded, generates an interrupt or DMA service request that, if enabled, signals the processor or DMA respectively to empty the Receive FIFO or to refill the Transmit FIFO.

The GSPI controller, as a host, provides the clock signal and controls the chip select line. Commands codes as well as data values are serially transferred on the data signals. The Processor asserts a chip select line to select the corresponding peripheral device with which it wants to communicate. The clock line is brought to the device whether it is selected or not. The clock serves as synchronization of the data communication.

35.1.2 DMA Controller

The GSPI controllers have an integrated DMA controller.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires that the peripheral control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires that the peripheral control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory. The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode.

Channel Control

- The source transfer width and destination transfer width are programmable. The width can be programmed to 1, 2, or 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. this number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual Channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. the block size is not limited by the source or destination transfer widths.
- Address incrementing modes: The DMA has a configurable mechanism for computing the source and destination addresses for the next transfer within the current block. The DMA supports incrementing addresses and constant addresses.

- Flexibility to configure any hardware handshake sideband interface to any of the DMA channels.
- Early termination of a transfer on a particular channel.

35.1.3 Reset

Each host controller has an independent rest associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into the corresponding reset register to bring the controller from reset state into operational mode.

35.1.4 Power Management

Device Power Down Support

In order to power down peripherals connected to the Processor GSPI bus, the idle configured state of the I/O signals must be retained to avoid transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when the bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. However, the GSPI bus architecture does not provide the architectural means to define dynamic latency tolerance messaging. Therefore, the interface supports this by reporting its service latency requirements to the platform power management controller via LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements. In this scheme, the latency requirement is a function of the controller state. The latency for transmitting data to/from its connected device at a given rate while the controller is active is representative of the active latency requirements. On the other hand if the device is not transmitting or receiving data and idle, there is no expectation for end to end latency.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end-to-end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

35.1.5 Interrupts

Each interface has the ability to interrupt and notify the driver that service is required

When an interrupt occurs, the device driver needs to read both the host controller and DMA interrupt status and transmit completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

35.1.6 Error Handling

Errors that might occur on the external GSPI signals are comprehended by the host controller and reported to the interface host controller driver through the MMIO registers.

35.2 Signal Description

Signal Name	Type	Description
GPP_E17/THC0_SPI1_CS#/ **GSPI0_CS0#**	O	**Generic SPI 0 Chip Select**
GPP_E11/THC0_SPI1_CLK/ **GSPI0_CLK**	O	**Generic SPI 0 Clock**
GPP_E13/THC_I2C0_SDA/ THC0_SPI1_IO1/**GSPI0_MISO**/ I2C4_SDA	I	**Generic SPI 0 MISO**
GPP_E12/THC_I2C0_SCL/ THC0_SPI1_IO0/**GSPI0_MOSI**/ I2C4_SCL	O	**Generic SPI 0 MOSI**
GPP_F17/THC1_SPI2_CS#/ ISH_SPIA_CS#/**GSPI1_CS0#**	O	**Generic SPI 1 Chip Select 0**
GPP_F11/THC1_SPI2_CLK/ ISH_SPIA_CLK/**GSPI1_CLK**	O	**Generic SPI 1 Clock**
GPP_F13/THC_I2C1_SDA/ I3C2_SDA/THC1_SPI2_IO1/ ISH_SPIA_MOSI/**GSPI1_MISO**/ I2C5_SDA	I	**Generic SPI 1 MISO**
GPP_F12/THC_I2C1_SCL/ I3C2_SCL/THC1_SPI2_IO0/ ISH_SPIA_MISO/**GSPI1_MOSI**/ I2C5_SCL	O	**Generic SPI 1 MOSI**
GPP_F18/THC1_INT#/ **GSPI0A_CS0#**	O	**Generic SPI 0A Chip Select**
GPP_F16/THC1_RST#/ **GSPI0A_CLK**	O	**Generic SPI 0A Clock**
GPP_F15/THC1_SPI2_IO3/ **GSPI0A_MISO**	I	**Generic SPI 0A MISO**
GPP_F14/THC1_SPI2_IO2/ **GSPI0A_MOSI**	O	**Generic SPI 0A MOSI**

35.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
GSPI0_MOSI	Pull Down	20 kohm± 30%	The integrated pull down is disabled after PROC_PWROK assertion
GSPI1_MOSI	Pull Down	20 kohm ± 30%	The integrated pull down is disabled after PROC_PWROK assertion
GSPI0_MISO	Pull Down	20 kohm ± 30%	
GSPI1_MISO	Pull Down	20 kohm ± 30%	

35.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
GSPI1_CS0#, GSPI0_CS0# , GSPI0A_CS0#	Primary	Undriven	Undriven	Undriven
GSPI1_CLK, GSPI0_CLK , GSPI0A_CLK	Primary	Undriven	Undriven	Undriven
GSPI1_MISO, GSPI0_MISO , GSPI0A_MISO	Primary	Undriven	Undriven	Undriven
GSPI1_MOSI, GSPI0_MOSI, GSPI0A_MOSI	Primary	Internal Pull-down	Driven Low	Internal Pull-down
Note: 1. Reset reference for primary well pins is RSMRST#.				

36.0 Touch Host Controller (THC)

Touch Host Controller provides a standard SPI interface for Processor to connect to external touch ICs. Only SPI IOs are supported.

THC also supports the GPIO based SPI interrupt from touch IC and supports hardware autonomous power management scheme within the Processor

Table 119. Acronyms

Acronyms	Description
CLK	Clock
CS	Chip Select
TPM	Trusted Platform Module

36.1 Functional Description

NOTE

THC-SPI and THC-I2C interface are not supported in Windows OS.

The Touch Host Controller (THC) supports a host controller interface to the touch IC for high bandwidth touch data transfer from SPI and I2C based touch ICs. THC provides low latency DMA services to the system memory for OS input stack.

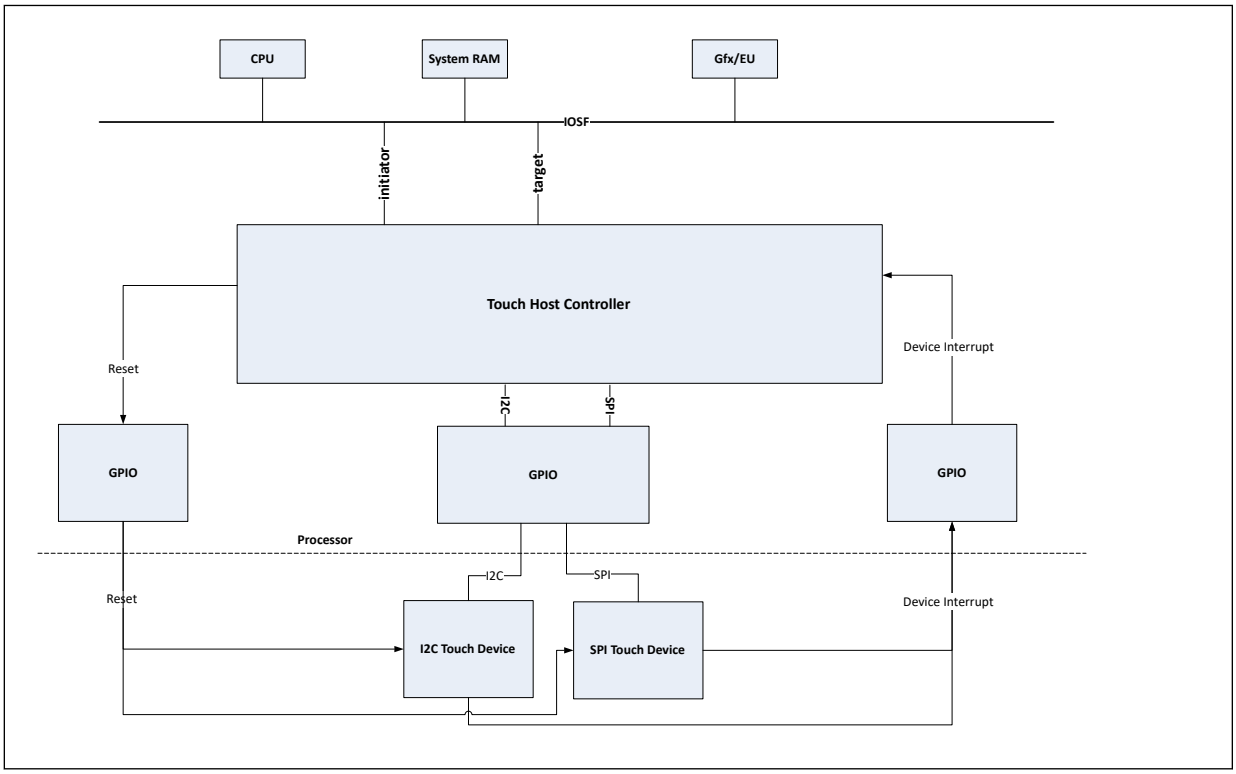
There are Two THC controllers available in Intel® Core™ Processor (Series 3) Processor Platform. Each can be configured to either I2C or SPI interface for the device connection. THC0 is the primary controller. If only one THC is used, THC0 is the preferred choice for ease of system FW/OS image integration.

- THC Controller
  - Touch Host controller supports SPI and I2C interface for touch ICs connection.
  - 1.8 V IO's supported for SPI and I2C interface
  - The THC-SPI supported frequency are are 8 MHz, 15 MHz, 17 MHz, 20 MHz, 25 MHz, 32 MHz, and 42 MHz.
  - Maximum frequency supported THC-I2C is 3.4 MHz.
  - Input for display sync signal for touch scroll smoothing

NOTE

Due to current THC panel availability, Intel validated up to 32 MHz Touch panels for THC-SPI interface even though the Maximum frequency supported is 42 MHz.

Figure 22. THC Block Diagram



36.2 Signal Description

Signal Name	Type	Description
GPP_E11/**THC0_SPI1_CLK**/GSPI0_CLK	O	**THC0_SPI1 Clock**: THC SPI1 clock output from Processor. Supports 42.67MHz.
GPP_F11/**THC1_SPI2_CLK**/ ISH_SPIA_CLK/GSPI1_CLK	O	**THC1_SPI2 Clock**: THC SPI2 clock output from Processor. Supports 42.67MHz.
GPP_E17/**THC0_SPI1_CS#**/GSPI0_CS0#		**THC0_SPI1 Chip Select**: Used to select the touch devices if it is connected to THC0_SPI1 interface.
GPP_F17/**THC1_SPI2_CS#**/ ISH_SPIA_CS#/GSPI1_CS0#	O	**THC1_SPI2 Chip Select**: Used to select the touch devices if it is connected to THC1_SPI2 interface.
GPP_E12/**THC_I2C0_SCL**/ **THC0_SPI1_IO0**/GSPI0_MOSI/ I2C4_SCL	I/O	**THC_I2C0_SCL**: THC I2C Link 0 Clock **THC0_SPI1_IO0**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E13/**THC_I2C0_SDA**/ **THC0_SPI1_IO1**/GSPI0_MISO/ I2C4_SDA	I/O	**THC_I2C0_SDA**: THC I2C Link 0 Data **THC0_SPI1_IO1**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E14/**THC0_SPI1_IO2**	I/O	**THC0_SPI1_IO2**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E15/**THC0_SPI1_IO3**	I/O	**THC0_SPI1_IO3**: A bidirectional signal used to support single, dual and quad mode data transfer.

continued...

Signal Name	Type	Description
GPP_F12/**THC_I2C1_SCL**/I3C2_SCL/ **THC1_SPI2_IO0**/ISH_SPIA_MISO/ GSPI1_MOSI/I2C5_SCL	I/O	**THC_I2C1_SCL**: THC I2C Link 1 Clock **THC1_SPI2_IO0**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F13/**THC_I2C1_SDA**/I3C2_SDA/ **THC1_SPI2_IO1**/ISH_SPIA_MOSI/ GSPI1_MISO/I2C5_SDA	I/O	**THC_I2C1_SDA**: THC I2C Link 1 Data **THC1_SPI2_IO1**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F14/**THC1_SPI2_IO2**/ GSPI0A_MOSI	I/O	**THC1_SPI2_IO2**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F15/**THC1_SPI2_IO3**/ GSPI0A_MISO	I/O	**THC1_SPI2_IO3**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E16/**THC0_RST#**	O	**THC0 Reset:** THC0_SPI1 Reset signal from Touch host controller.
GPP_F16/**THC1_RST#**/GSPI0A_CLK	O	**THC1 Reset:** THC1_SPI2 Reset signal from Touch host controller.
GPP_E18/**THC0_INT#**	I	**THC0 interrupt:** THC0_SPI1 Interrupt signal.
GPP_F18/**THC1_INT#**/GSPI0A_CS0#	I	**THC1 interrupt:** THC1_SPI2 Interrupt signal.
GPP_E22/**THC0_DSYNC**	I	**THC0 DSYNC**: THC0-SPI Port1 External Display Frame Sync
GPP_F22/**THC1_DSYNC**/ISH_GP8A	I	**THC1 DSYNC**: THC1-SPI Port2 External Display Frame Sync

36.3 Integrated Pull-Ups and Pull-Downs

None

36.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
THC0_SPI1_CLK	Primary	Undriven	Undriven	Undriven
THC1_SPI2_CLK	Primary	Undriven	Undriven	Undriven
THC0_SPI1_CS#	Primary	Undriven	Undriven	Undriven
THC1_SPI2_CS#	Primary	Undriven	Undriven	Undriven
THC0_SPI1_IO[0:3]	Primary	Undriven	Undriven	Undriven
THC1_SPI2_IO[0:3]	Primary	Undriven	Undriven	Undriven
THC0_RST#	Primary	Undriven	Undriven	Undriven
THC1_RST#	Primary	Undriven	Undriven	Undriven
THC0_INT#	Primary	Undriven	Undriven	Undriven
THC1_INT#	Primary	Undriven	Undriven	Undriven
THC0_DSYNC	Primary	Undriven	Undriven	Undriven
THC1_DSYNC	Primary	Undriven	Undriven	Undriven
THC_I2C0_SCL	Primary	Undriven	Undriven	Undriven
THC_I2C0_SDA	Primary	Undriven	Undriven	Undriven

continued...

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
THC_I2C1_SCL	Primary	Undriven	Undriven	Undriven
THC_I2C1_SDA	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

37.0 Intel® Serial IO Universal Asynchronous ReceiverTransmitter (UART) Controlle

The Processor implements three independent UART interfaces, UART0, UART1 and UART2. Each UART interface is a 4-wire interface supporting up to 6.25 Mbit/s.

The interfaces can be used in the low-speed, full-speed, and high-speed modes. The UART communicates with serial data ports that conform to the RS-232 interface protocol.

UART2 only implements the UART Host controller and does not incorporate a DMA controller which is implemented for UART0 and UART1. Therefore, UART2 is restricted to operate in PIO mode only.

The UART interfaces support the following features:

- Up to 6.25 Mbit/s Auto Flow Control mode as specified in the 16750 standards
- Transmitter Holding Register Empty (THRE) interrupt mode
- 64-byte TX and 64-byte RX host controller FIFOs
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- Functionality based on the 16550 industry standards
- Programmable character properties, such as number of data bits per character (5-8), optional parity bit (with odd or even select) and number of stop bits (1, 1.5, or 2)
- Line break generation and detection
- DMA signaling with two programmable modes
- Prioritized interrupt identification
- Programmable FIFO enable/disable
- Programmable serial data baud rate
- Modem and status lines are independently controlled
- Programmable BAUD RATE supported (baud rate = (serial clock frequency)/ (16xdivisor))

NOTES

1. SIR mode is not supported.
2. External read enable signal for RAM wake up when using external RAMs is not supported.

Table 120. Acronyms

Acronyms	Description
DMA	Direct Memory Access
UART	Universal Asynchronous Receiver/Transmitter
LSx	Low speed IO Controller

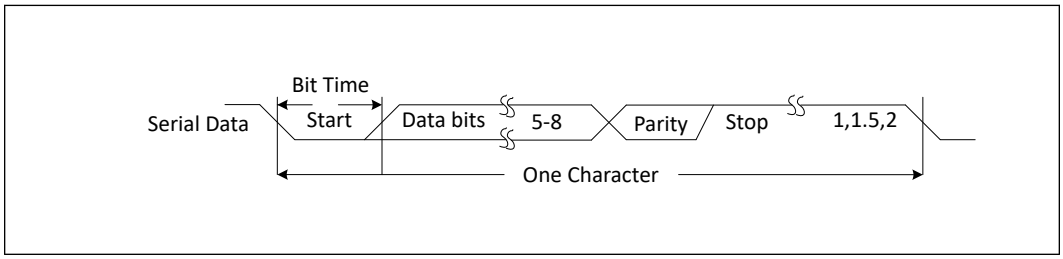
37.1 Functional Description

37.1.1 UART Serial (RS-232) Protocols Overview

Because the serial communication between the UART host controller and the selected device is asynchronous, Start and Stop bits are used on the serial data to synchronize the two devices. The structure of serial data accompanied by Start and Stop bits is referred to as a character.

An additional parity bit may be added to the serial character. This bit appears after the last data bit and before the stop bit(s) in the character structure to provide the UART Host Controller with the ability to perform simple error checking on the received data.

Figure 23. UART Serial Protocol



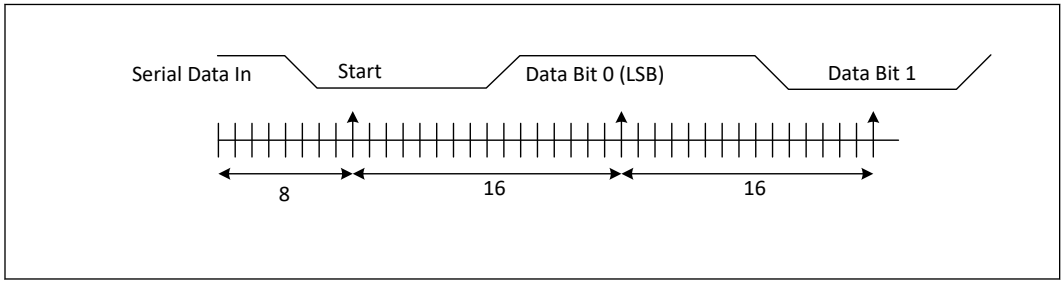
The UART Host Controller Line Control Register (LCR) is used to control the serial character characteristics. The individual bits of the data word are sent after the Start bit, starting with the least significant bit (LSB). These are followed by the optional parity bit, followed by the Stop bit(s), which can be 1, 1.5, or 2.

The Stop bit duration implemented by UART host controller may appear longer due to idle time inserted between characters for some configurations and baud clock divisor values in the transmit direction.

All bit in the transmission (with exception to the half stop bit when 1.5 stop bits are used) are transmitted for exactly the same time duration (which is referred to as Bit Period or Bit Time). One Bit Time equals to 16 baud clocks.

To ensure stability on the line, the receiver samples the serial input data at approximately the midpoint of the Bit Time once the start bit has been detected.

Figure 24. UART Receiver Serial Data Sample Points



37.1.2 16550 8-bit Addressing - Debug Driver Compatibility

NOTE

The Processor UART host controller is not compatible with legacy UART 16550 debug-port drivers. The UART host controller operates in 32-bit addressing mode only. UART 16550 legacy drivers only operate with 8-bit addressing. In order to provide compatibility with standard in-box legacy UART drivers a 16550 Legacy Driver mode has been implemented in the UART controller that will convert 8-bit addressed accesses from the 16550 legacy driver to the 32-bit addressing that the UART host controller supports. The UART 16550 8-bit Legacy mode only operates with PIO transactions. DMA transactions are not supported in this mode.

37.1.3 DMA Controller

The UART controllers 0 and 1 (UART0 and UART1) have an integrated DMA controller. Each channel contains a 64-byte FIFO. Max. burst size supported is 32 bytes.

UART controller 2 (UART2) only implements the host controllers and does not incorporat is not bee a DMA. Therefore, UART2 is restricted to operate in PIO mode only.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires that the peripheral control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires that the peripheral control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory (such as DDR or SRAM). The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode

Channel Control

- The source transfer width and destination transfer width are programmable. It can vary to 1 byte, 2 bytes, and 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. this number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual Channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. Block size is not limited by the source or destination transfer widths.
- Address incrementing modes: The DMA has a configurable mechanism for computing the source and destination addresses for the next transfer within the current block. The DMA supports incrementing addresses and constant addresses.
- Flexibility to configure any hardware handshake sideband interface to any of the DMA channels.
- Early termination of a transfer on a particular channel.

37.1.4 Reset

Each host controller has an independent rest associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered off and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

37.1.5 Power Management

Device Power Down Support

In order to power down peripherals connected to the processorUART bus, the idle, configured state of the I/O signals must be retained to avoid transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when the bus is powered off (power gated). The processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The UART bus architecture, however, does not provide the architectural means to define dynamic latency tolerance messaging. Therefore, the interface supports this by reporting its service latency requirements to the platform power management controller via LTR registers.

The controller’s latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements. In this scheme, the latency requirement is a function of the controller state. The latency for transmitting data to/from its connected device at a given rate while the controller is active is representative of the active latency requirements. On the other hand if the device is not transmitting or receiving data and idle, there is no expectation for end to end latency.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

37.1.6 Interrupts

UART interface has the ability to interrupt and notify the driver that service is required

When an interrupt occurs, the device driver needs to read both the host controller and DMA status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

37.1.7 Error Handling

Errors that might occur on the external UART signals are comprehended by the host controller and reported to the interface host controller driver through the MMIO registers.

37.2 Signal Description

Signal Name	Type	Description
GPP_H08/**UART0_RXD**	I	**UART 0 Receive Data**
GPP_H09/**UART0_TXD**	O	**UART 0 Transmit Data**
GPP_H10/**UART0_RTS#**/ I3C1A_SDA/ISH_GP10A	O	**UART 0 Request to Send**
GPP_H11/**UART0_CTS#**/ I3C1A_SCL/ISH_GP11A	I	**UART 0 Clear to Send**
GPP_H06/I2C3_SDA/ **UART1_RXD**/ ISH_UART1A_RXD	I	**UART 1 Receive Data**
GPP_H07/I2C3_SCL/ **UART1_TXD**/ ISH_UART1A_TXD	O	**UART 1 Transmit Data**
GPP_H14/ ISH_UART1_RXD/ **UART1A_RXD**/ ISH_I2C1_SDA/ ISH_I3C1_SDA	O	**UART 1A Receive Data**

continued...

Signal Name	Type	Description
GPP_H15/ ISH_UART1_TXD/ **UART1A_TXD**/ ISH_I2C1_SCL/ ISH_I3C1_SCL	I	**UART 1A Transmit Data**
GPP_F01/CNV_BRI_RSP/ **UART2_RXD**	I	**UART 2 Receive Data**
GPP_F02/CNV_RGI_DT/ **UART2_TXD**	O	**UART 2 Transmit Data**
GPP_F00/CNV_BRI_DT/ **UART2_RTS#**	O	**UART 2 Request to Send**
GPP_F03/CNV_RGI_RSP/ **UART2_CTS#**	I	**UART 2 Clear to Send**

37.3 Integrated Pull-Ups and Pull-Downs

None.

37.4 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
UART[2:0]_RXD	Primary	Undriven	Undriven	Undriven
UART[2:0]_TXD	Primary	Undriven	Undriven	Undriven
UART2_RTS# **UART0_RTS#**	Primary	Undriven	Undriven	Undriven
UART2_CTS# **UART0_CTS#**	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

37.5 LSx

LSx interface supports Four ports. Each port of the LSx controller has two bi-directional signals configured either as Tx (Output) or Rx (Input). Operating voltage of the LSx interface is 1.8 V. LSx controller is responsible for link initialization/ management of HSIO in the Thunderbolt subsystem.

37.5.1 LSx Signal Description

Signal Name	Type	Description
GPP_C17/ **TBT_LSX0_RXD**/ DDP0_CTRLDATA	I	LSx 0 Receive Data
GPP_C16/ **TBT_LSX0_TXD**/ DDP0_CTRLCLK	O	LSx 0 Transmit Data

continued...

Signal Name	Type	Description
GPP_D19/**TBT_LSX0_OE**	O	Controls the direction of external level shifter to support TBT3 active cables with bidirectional communication.
GPP_C19/ **TBT_LSX1_RXD**/ DDP1_CTRLDATA	I	LSx 1 Receive Data
GPP_C18/ **TBT_LSX1_TXD**/ DDP1_CTRLCLK	O	LSx 1 Transmit Data
GPP_B16/**TBT_LSX1_OE**	O	Controls the direction of external level shifter to support TBT3 active cables with bidirectional communication.

37.5.2 Integrated Pull-Ups and Pull-Downs

None.

37.5.3 IO Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
TBT_LSX[0:1]_RXD	Primary	Undriven	Undriven	Undriven
TBT_LSX[0:1]_TXD	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

38.0 Private Configuration Space Port ID

The Processor incorporates a wide variety of devices and functions. The registers within these devices are mainly accessed through the primary interface, such as PCI configuration space and IO/MMIO space. Some devices also have registers that are distributed within the Processor Private Configuration Space at individual endpoints (Target Port IDs) which are only accessible through the Processor Sideband Interface. These Processor Private Configuration Space Registers can be addressed via SBREG_BAR or through SBI Index Data pair programming. .

Table 121. Private Configuration Space Register Target Port IDs

Processor Device/Function Type	Target Port ID (hex)
General Purpose I/O (GPIO) Community 0	F259h
General Purpose I/O (GPIO) Community 1	F25Ah
General Purpose I/O (GPIO) Community 3	F25Bh
General Purpose I/O (GPIO) Community 4	F25Ch
General Purpose I/O (GPIO) Community 5	F25Dh
PCIe Controller #1 (SPA)	F201h
PCIe Controller #3 (SPC)	F203h
SMBus	F26Bh
eSPI / SPI	F26Dh
xHCI	F209h
CNVi	F251h
PSF4	F2B0h
PSF6	F2B1h
PSF8	F2B2h
PSF14	F2B3h
PSF15	F2B4h
ISH Controller	F25Eh
USB 2.0	F222h
UART, I^2C, GSPI, I^3C	F265h
Integrated Clock Controller (ICC)	F273h
GbE	F20Dh
Real Time Clock (Host)	F26Ch
LSx	F267h
UFS	F20Eh

39.0 Testability and Monitoring

The processor provides a comprehensive set of testability and monitoring features to support board and system-level validation, debug, and telemetry. These include standard JTAG boundary scan, dedicated debug and run-control signals, and PMT for hardware telemetry.

39.1 Signal Description

Table 122. Testability Signals

Signal Name	Type	Description
DBG_PMODE	O	ITP Power Mode Indicator. This signal is used to transmit processor and power/reset information to the Debugger.
PRDY#	O	**Probe Mode Ready:** PRDY# is a processor output used by debug tools to determine processor debug readiness
PREQ#	IOD	**Probe Mode Request:** PREQ# is used by debug tools to request debug operation of the processor.
SOC JTAG Signals		
SOC_JTAG_TCK	I/O	**Test Clock Input (TCK):** The test clock input provides the clock for the JTAG test logic.
SOC_JTAG_TMS	IOD	**Test Mode Select (TMS):** The signal is decoded by the Test Access Port (TAP) controller to control test operations.
SOC_JTAG_TDI	IOD	**Test Data Input (TDI):** Serial test instructions and data are received by the test logic at TDI.
SOC_JTAG_TDO	IOD	**Test Data Output (TDO):** TDO is the serial output for test instructions and data from the test logic defined in this standard.
SOC_JTAG_TRST#	I/O	**Test Reset(TRST):** Resets the Test Access Port (TAP) logic. This signal should be driven low during power on Reset.
Breakpoint and Performance Monitor Signals		
BPM[0]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
BPM[1]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
Boot Halt Signal		
BOOTHALT#	IOD	**Boot Halt :** This signal is used for platform boot halt.