Intel® Pentium® Silver and Intel® Celeron® Processors

Datasheet, Volume 1 of 2

Rev. 006

December 2022

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

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

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

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

Intel technologies’ features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

Contents

Figures

Tables

Revision History

Document Number	Revision Number	Description	Revision Date
633935	001	Initial Release	January 2021
	002	Storage on page 182 • Updated the signal description of **EMMC_RESET_N** in eMMC Signals Description on page 182	February 2021
	003	Power Management on page 33 • **Updated** G3 value for voltage rail VCCRTC in Table 19 on page 53	April 2021
	004	Memory on page 65 • **Added** System Memory Timing Support on page 67	April 2021
	005	CPU Electrical Specifications on page 136 • **Updated** the minimum and maximum values of **Tco CPU clock to data delay** in SVID AC Specifications on page 138	June 2021
	006	Power Management on page 33 • **Updated** Table 32 on page 78 • **Updated** description for PMC_ALERT_N and PMC_SYS_RESET_N in Table 12 on page 45 System Management Interface and SMLink on page 238 • **Updated** description for PMC_ALERT_N signal in Signal Description on page 238	December 2022

1.0 Introduction

Intel® Pentium® Silver and Intel® Celeron® Processor is an Intel Architecture (IA) Multi-Chip Processor (MCP) 2-Chip Package, built on a 10 nm CPU and a 14 nm Platform Controller Hub (PCH) into a single package. Both dies are connected via the On Package Interface (OPI).

This document is intended for Original Equipment Manufacturers (OEMs), Original Design Manufacturers (ODM) and BIOS vendors.

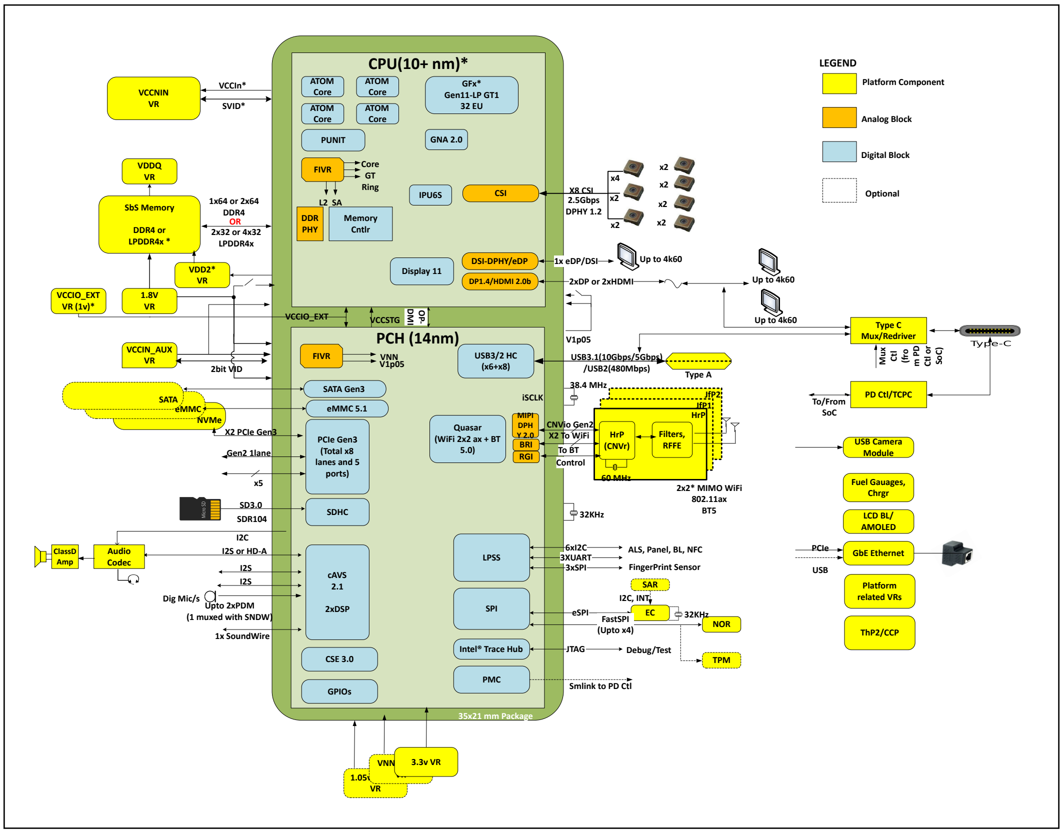
NOTE

Throughout this document, the name **SoC** is used as a general term and refers to all Intel® Pentium® Silver and Intel® Celeron® Processor family SKUs.

This manual assumes a working knowledge of the vocabulary and principles of interfaces and architectures such as PCI Express* (PCIe*), Universal Serial Bus (USB), Advance Host Controller Interface (AHCI), eXtensible Host Controller Interface (xHCI), and so on.

1.1 Block Diagram

Figure 1. Platform Block Diagram



1.2 CPU Core Overview

Category	Feature Description
CPU Cores	Quad/Dual IA ATOM Core • 3-way Superscalar, Out of Order Execution (OOE) • 10 nm CPU technology
Modules/Caches	• 1 set of 2 cores (for Dual Core) or 2 set of 2 cores (for Quad Core) • On-die, parity protected 32KB 8-way (64 sets) L1 instruction cache and 32KB 8-way (64 sets) L1 data cache per core • On-die, 12-way (2048 sets) L2 unified cache for all cores(module)
Architecture	Intel® 64-bit
Virtualization Architecture	Intel® Virtualization Technology • VTx-2 with Extended Page Table • VT-d
Burst Technology	1/2/3/4 Core Burst Technology • All cores in C0 state runs at the same frequency

continued...

Category	Feature Description
Thermal Management	Supported by means of Intel® Thermal Monitor (TM1 and TM2)
Power Management	• Enhanced Intel SpeedStep® Technology and Intel® Speed Shift Technology • Core C-States: C0, C1, C1E, C6, C6S, C7,C8,C10 • Module C-States: MC0, MC6
Other features	Security Technologies: • Branch Monitoring Counters, Intel® AES-NI, PCLMULQD, Execute Disable Bit, Boot Guard, Intel® SMEP, Intel® SMAP, Intel® SHA Extensions, User Mode Instruction Prevention, Read Processor ID. Power and Performance Technologies: • x2APIC, Cache Line Write Back Debug Technologies: • Intel® Processor Trace

1.3 Features Supported

Component	Category	MCP
CPU	Number of Cores	Up to four Tremont Atom Cores
	Burst Speed	Dependent on number of active cores and CPU SKU
	LFM/HFM	400 MHz / 1.1-2 GHz
	L1 Cache	32 KB Instruction, 32 KB Data per core
	L2 Cache	1.5 MB
	L3 Cache	4 MB
Package	Type	Type 3 FCBGA 35 mm x 24 mm
	I/O Count	720
	Pin Count	1338
	Minimum Ball Pitch	0.593 mm
	Z-Height	1.406 mm
	TDP	6 W (Mobile) / 10 W (DT)
	Operating Temperature	0 ºC to +85 ºC ambient (TjMax 105degC)
Graphics	Gen	Gen11 LP
	LFM/HFM/Burst	200 MHz/≤ 450MHz/≤900MHz
	Execution Units	Up to 32 in a 1x4x8 configuration [16EU (1x2x8) and 24EU (1x4x6) configurations are also supported]
Display	Gen	Gen11 3x Pipe
	DDI 0	eDP / MIPI DSI 4L / DP / HDMI
	DDI 1	MIPI DSI 4L / DP / HDMI
	DDI 2	DP / HDMI
	Display Serial Interface (DSI)	MIPI-DSI 1.2 @ 2.5 Gbps

continued...

Component	Category	MCP
	Embedded Display Port (eDP*)	eDP 1.4b @ 5.4 Gbps
	Display Port (DP)	DP 1.4a @ 8.1 Gbps [HBR3 is supported with DP alternate mode over type-C,HBR2 is natively supported over DP connector. HBR3 over DP connector is supported only with retimer]
	High Definition Multimedia Interface (HDMI)	HDMI 2.0b @ 5.94 Gbps (With Platform Level Shifters above HD resolution)
	Maximum Resolution	4K2K
Media	Decode Codec	VP9 or HEVC 8b/10b 4:4:4/4:2:0 4k30/4k60
	Encode Codec	FF-VP9 or FF-HEVC 10b 4:4:4/4:2:0 4k30/4k60
Memory	Maximum Size Supported	2x32 (1 channel) or 4x32 (2 channel) LPDDR4x 1x64 (either of the 2 channels) and 2x64 DDR4

	Supported Transfer Data Rates (MT/s)	LPDDR4x = 2933 MT/s
		DDR4 = 2933 MT/s

Imaging	Number of Lanes	Eight lanes of DPHY1.2
	Number of Cameras	Four Total Cameras / Three Concurrent cameras
	Supported Data Rate (Total BW/lane)	2.5 Gbps/lane
Audio	Audio endpoint connections	Two HDA One Intel Integrated Display Audio Codec (HDMI / DP) Three I2S / PCM codec Two PDM DMIC module One SoundWire* segment
	SDO/SDI	Up to 48 Mb/s / 24 Mb/s
	Codec Support	44.1 kHz sampling rate up to 24 MHz BCLK
	Audio Engine	Dual Core Tensilica core with HIFI 3 Audio Engine @ 400 MHz [1 MB L2 high performance SRAM and 64 KB L2 low power SRAM]
	Speech Accelerator	GNA 2.0
USB	Number of Ports	Up to eight USB ports
	Type-C	External Type-C solution only
	USB 3.2 2x1 SuperSpeed Port	Two
	Maximum USB 3.2 2x1 Speed	10 Gb/s
	USB 3.2 1x1 Port	Four
	Maximum USB 3.2 1x1 Speed	5 Gb/s
	USB 2.0 Port	Eight
	Maximum USB 2.0 Speed	480 Mb/s
PCIe* Gen3	Ports	Up to five Ports Eight Lanes (multiplexed)
	Maximum Speed	8 GT/s

continued...

Component	Category	MCP
Storage	eMMC*	5.1
	Maximum eMMC* Speed	400 MB/s
	Sata	2x Gen3 ports
	Maximum Speed	Gen 3 (6.0Gb/s)
	Secure Digital	SD 3.0
	SD Speed	Default Mode: Up to 12.5 MB/s High Speed Mode: Up to 25 MB/s UHS-I Mode: Up to 100 MB/s
LPSS	SPI	Three
	SPI Speed	25 Mbps (Master Mode Only)
	UART	Three
	UART Speed	3.8 Mbps
	I2C	Six
	I2C Speed	3.4 Mbps (Master Mode Only)
SMBus	Ports	One
	Maximum Speed	100 kHz
Fast SPI	Controller	Controller: One Devices supported: Three (two for Flash, one for TPM) (FST_SPI supports up to three loads)
	Maximum Fast SPI Frequency	50 MHz
eSPI	Maximum eSPI Frequency	60 MHz
Clocks	38.4 MHz, 32 kHz Xtal Inputs	
Interrupt	2 X Interrupt Controller (8256 and I/O APIC)	15 Interrupts, Message Signaled Interrupts (MSI) Support
Timer		8x HPET Intel® 8254 timer
RTC	256 byte Battery backed RAM	
ACPI	Advanced Configuration Power Interface (ACPI) 5.0a Compliant Power Management	
Security	Technology	Trusted Platform Module (TPM2.0), Intel® Platform Trust Technology
Wireless Connectivity	Wi-Fi	Integrated CNVi, 802.11ax [Wi-Fi 6] (1x1 and 2x2)
	Bluetooth*	Integrated CNVi BT 5.x using UART/I2S/USB2

1.4 SKUs

Table 1. SKU MAP

Intel® Pentium® Silver and Intel® Celeron® Processor	N6000	N5100	N4500	N6005	N5105	N4505
Proposed Branding	Pentium	Celeron	Celeron	Pentium	Celeron	Celeron
Use Condition	PC Client	PC Client	PC Client	PC Client	PC Client	PC Client
Cores	4	4	2	4	4	2
TDP	6 W	6 W	6 W	10 W	10 W	10 W
CPU HFM	1.1 GHz	1.1 GHz	1.1 GHz	2.0 GHz	2.0 GHz	2.0 GHz
CPU Max Burst Frequency	3.3 GHz	2.8 GHz	2.8 GHz	3.3 GHz	2.9 GHz	2.9 GHz
Gen 11LP	32EUs	24EUs	16EUs	32EUs	24EUs	16EUs
GFX HFM	350 MHz	350 MHz	350 MHz	450 MHz	450 MHz	450 MHz
GFX burst Frequency	850 MHz	800 MHz	750 MHz	900 MHz	800 MHz	750 MHz
TJ	0 to 105 °C	0 to 105 °C	0 to 105 °C	0 to 105 °C	0 to 105 °C	0 to 105 °C

2.0 Technologies

This chapter provides a high-level description of Intel technologies implemented in the processor.

The implementation of the features may vary between the processor SKUs.

Details on the different technologies of Intel processors and other relevant external notes are located at the Intel technology web site:http://www.intel.com/technology/.

Details on the Intel® Architecture Instruction Set Extensions and Future Features and Programming Reference is found here: https://software.intel.com/sites/default/files/managed/c5/15/architecture-instruction-set-extensions-programming-reference.pdf

2.1 Security Technologies

This section contains information about the following:

- Branch Monitoring Counters
- Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)
- Perform Carry-Less Multiplication Quad word (PCLMULQDQ) Instruction
- Intel® Secure Key
- Execute Disable Bit
- Boot Guard Technology
- Intel® Supervisor Mode Execution Protection (SMEP)
- Intel® Supervisor Mode Access Protection (SMAP)
- Intel® Secure Hash Algorithm Extensions (Intel® SHA Extensions)
- User Mode Instruction Prevention (UMIP)
- Read Processor ID (RDPID)

2.1.1 Branch Monitoring Counters

Branch monitoring technology allows monitor and detection a set of heuristics within an execution window in a program. This heuristics can be used for detecting abnormal behavior in code execution and signal the anti-malware software of its occurrence.

These technology allows such Anti-Virus software to receive a signal (interrupt) when a counter threshold has been reached. Branch Monitoring allows software to perform non-intrusive runtime analysis of ROP (Return Oriented Programming) attacks on applications.

The heuristics are based on certain performance monitoring statistics, measured dynamically over a short configurable window period. Anti-malware software has the responsibility to configure the Hardware statistics of interest and the Window size via MSR registers. Anti-malware SW is also responsible for post-processing any signaled event due to a detection condition. Such signaling is not considered 100% reliable and thus the anti-malware software is the ultimate decision maker to avoid false positives, while at the same time maintaining sufficient sensitivity for detecting malware.

2.1.2 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 are 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 industry applications, and is widely deployed in various protocols.

Intel® AES-NI consists of six 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.

This generation of the processor has increased the performance of the Intel® AES-NI significantly compared to previous products.

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.

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

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

2.1.4 Intel® Secure Key

The processor supports Intel® Secure Key (formerly known as Digital Random Number Generator (DRNG), a software visible random number generation mechanism supported by a high quality entropy source. This capability is available to programmers through the RDRAND instruction. 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 instruction include cryptographic key generation as used in a variety of applications, including communication, digital signatures, secure storage, etc.

RDRAND 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

2.1.5 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 utilize buffer overrun vulnerabilities and can, thus, help improve the overall security of the system.

2.1.6 Boot Guard Technology

Boot Guard technology is a part of boot integrity protection technology. Boot Guard can help protect the platform boot integrity by preventing execution of unauthorized boot blocks. With Boot Guard, platform manufacturers can create boot policies such that invocation of an unauthorized (or untrusted) boot block will trigger the platform protection as per the manufacturer's defined policy.

With verification based in the hardware, Boot Guard extends the trust boundary of the platform boot process down to the hardware level.

Boot Guard accomplishes this by:

- Providing of hardware-based Static Root of Trust for Measurement (S-RTM) and the Root of Trust for Verification (RTV) using Intel architectural components.
- Providing of architectural definition for platform manufacturer Boot Policy.
- Enforcing of manufacture provided Boot Policy using Intel architectural components.

Benefits of this protection is that Boot Guard can help maintain platform integrity by preventing re-purposing of the manufacturer's hardware to run an unauthorized software stack.

NOTE

Boot Guard availability may vary between the different SKUs.

2.1.7 Intel® Supervisor Mode Execution Protection (SMEP)

Intel® Supervisor Mode Execution Protection (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

2.1.8 Intel® Supervisor Mode Access Protection (SMAP)

Intel® Supervisor Mode Access Protection (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 the *Intel ® 64 Architectures Software Developer's Manual, Volume 3*: http://www.intel.com/products/processor/manuals

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

Intel® Secure Hash Algorithm (SHA) is one of the most commonly employed cryptographic algorithms. Primary usages of Intel® SHA include data integrity, message authentication, digital signatures, and data de-duplication. As the pervasive use of security solutions continues to grow, Intel® 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 Intel® SHA in our everyday computing devices, the new instructions are designed to provide a needed boost of performance to hashing a single buffer of data. The performance benefits not only help improve responsiveness and lower power consumption for a given application, they may enable developers to adopt Intel® 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. More information on Intel® SHA is available at http://software.intel.com/en-us/articles/intel-sha-extensions.

2.1.10 User Mode Instruction Prevention (UMIP)

User Mode Instruction Prevention (UMIP) provides additional hardening capability to 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 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

2.1.11 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

2.2 Power and Performance Technologies

This section contains information about the following:

- Intel® Smart Cache Technology
- IA Core Level 1 and Level 2 Caches
- Enhanced Intel SpeedStep® Technology
- Intel® Speed Shift Technology
- Intel® 64 Architecture x2APIC
- Cache Line Write Back (CLWB)
- Converged Audio Voice Speech (cAVS)
- Integrated Connectivity (CNVi)

2.2.1 Intel® Smart Cache Technology

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

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

The LLC is shared between all IA cores as well as the Processor Graphics. Also, it has an additional 1280 KB L3 cache dedicated to the Graphics.

The first level cache is not shared between physical cores and each physical core has a separate level 1 cache. The second level caches is shared between all physical cores.

The size of the LLC is 4 MB and is a 16 way associative cache. It is parity protected.

2.2.2 IA Core Level 1 and Level 2 Caches

The first level cache is divided into a data cache and an instruction cache. The processor first level cache size is 32 KB for data and 32 KB for instructions. The first level cache is an eight way associative cache and is parity protected.

The second level cache holds both data and instructions. The L2 cache size is 1.5 MB and is a 12 way associative cache. It is shared across the four cores in the module and is ECC protected (1-bit correct and 2-bits detect).

2.2.3 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 frequency 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 processor IA cores.
  - Once the voltage is established, the PLL locks on to the target frequency.
  - All active processor IA cores share the same frequency and voltage. In a multi-core processor, the highest frequency P-state requested among all active IA 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. All of Processor Cores must be in the same P-state at any given time.

2.2.4 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 request a 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.

2.2.5 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 performance of interrupt delivery.
- Reduces 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 x2APIC mode, APIC registers are accessed through 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, a new operating system and a new BIOS are both needed, with special support for x2APIC mode.
- The x2APIC architecture provides backward compatibility to the xAPIC architecture and forward extensible for future Intel platform innovations.

NOTE

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

2.2.6 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 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 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/c5/15/architecture-instruction-set-extensions-programming-reference.pdf

2.2.7 Converged Audio Voice Speech (cAVS)

Converged Audio Voice Speech (cAVS) 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 CPU and/or DSP providing the audio enrichment. It may also be used as a host based sensor hub for managing various context information on the platform.

NOTE

For more information, refer to Features Supported on page 17 and Audio, Voice, and Speech on page 150.

2.2.8 Integrated Connectivity (CNVi)

Integrated Connectivity (CNVi) is a general term referring to a family of connectivity solutions.

Below is the summary of the CNVi technology. For more information, refer to Features Supported on page 17 and Connectivity Integrated (CNVi) on page 157.

- Digital Integration of Wireless connectivity functionality
  - Wi-Fi* and Bluetooth*
  - Support 802.11ac 2x2 160 MHz bandwidth
  - Support 802.11ax 2x2 160 MHz bandwidth
  - Support Bluetooth 4.2, 5.0
- BT support via USB
- Support connectivity processing subsystems for:
  - Wi-Fi*, as a PCIE device through connection to internal fabric.
  - Bluetooth*, with USB2 connection. USB2 is connected on-die to XHCI controller thru internal UTMI bridge.
- Multi-Function UART (MFUART), with one UART connected on-die to Programmable Services Engine and two UART having direct connection to GPIO pins for WiGig and LTE co-existence.

- Support Hammock Harbor time synchronization for Wi-Fi* only, no support for BT.
- Support swappable M.2 for integrated or discrete CNV module.

2.3 Debug Technologies

This section contains information about the following:

- Intel® Processor Trace
- JTAG
- Intel® Trace Hub
- Intel® Direct Connect Interface (DCI)

2.3.1 Intel® Processor Trace

Intel® Processor Trace (Intel® PT) is a tracing capability added to Intel® Architecture, for use in software debug and profiling. Intel® PT provides the capability for precise software control flow and timing information, with limited impact to software execution. This provides enhanced ability to debug software crashes, hangs, or other anomalies, as well as responsiveness and short-duration performance issues.

Refer to the Intel® 64 Architectures Software Developer's Manual, for more information. Available at:http://www.intel.com/products/processor/manuals.

2.3.2 JTAG

This section contains information regarding the testability signals that provides access to JTAG, Run-control, system control, and observation resources. JTAG (TAP) port is compatible with the IEEE Standard Test Access Port and Boundary Scan Architecture 1149.1 and 1149.6 Specification, as detailed per device in each BSDL file. JTAG Pin definitions are from IEEE Standard Test Access Port and Boundary Scan Architecture (IEEE Std. 1149.1-2013). MIPI-60 Debug Port (Connector) provides access to JTAG Port. JTAG may also be accessible via Intel® DCI for closed chassis debug usage.

2.3.3 Intel® Trace Hub

Intel® Trace Hub is a debug architecture that unifies hardware and software system visibility. Intel® Trace Hub is not merely intended for hardware debug or software debug, but full system debug. This includes debugging hardware and software as they interact and produce complex system behavior. Intel® Trace Hub defines new features and also leverages some existing debug technologies to provide a complete framework for hardware and software co-debug, software development and tuning, as well as overall system performance optimization.

Intel® Trace Hub is a set of silicon features with supported software API. The primary purpose is to collect trace data from different sources in the system and combine them into a single output stream with time-correlated to each other. Intel® Trace Hub uses common hardware interface for collecting time-correlated system traces through standard destinations (refer list below). Intel® Trace Hub adopts industry standard (MIPI* STPv2) debug methodology for system debug and software development.

There are multiple trace sources planned to be supported in the platform.

- BIOS

- Architecture Event Trace (AET)
- Power Management Event Trace
- Hardware Signals
- SW and FW Traces
- Windows* ETW (for driver or application)

There are multiple destinations to receive trace data from the Intel® Trace Hub:

- Intel® Direct Connect Interface (Intel® DCI)
  - — DCI.OOB
  - — DCI.USB2/3
- System Memory
- PTI (CFG)

2.3.4 Intel® Direct Connect Interface (DCI)

Intel® Direct Connect Interface (DCI) is an Intel® tool transport technology that allows connection between the host system running a debug, validation or test tool stack and a closed chassis debug through of USB 3.1 Gen2 and USB 2.0 Host ports out from Intel® processor. Intel® DCI requires embedded in the silicon to "bridge" the gap between standard I/O ports and the debug interfaces including JTAG, probe mode, hooks, trace infrastructure etc. To control the operation of this embedded logic, Intel® DCI packet based protocol is invented to allow controls and data can be sent or received. This protocol can operate over a few different physical transport paths to the target which known as "hosting interfaces".

NOTE

Further information on how to run Intel® DCI through Intel® System Studio System software is available at, https://software.intel.com/en-us/articles/system-debugging-via-direct-connect-interfacedci-of-intel-system-debug.

2.4 Intel® Debug and Tool

This section contains information about the following:

- Open Chassis Debug
- Closed Chassis Debug

2.4.1 Open Chassis Debug

MIPI-60 Debug Port is available for Open Chassis Debug access. The MIPI-60 JTAG topology is "Merged-Parallel", the CPU and PCH JTAG signals share common connector pins, except for TCKs, for which there are two dedicated pins. It consists of JTAG (TAP), CPU Run-control, PTI CFG (To correlate and be consistent as mentioned in Intel® Trace Hub on page 29) and other miscellaneous signals such as DFx hard-straps, system status, triggers in/out, reset generation, power controls. Lauterbach and TRACE32 are the debug tool set supported via MIPI-60 Debug Port. The TRACE 32 debugger allows to test embedded hardware and software.

2.4.2 Closed Chassis Debug

Closed Chassis Debug is a means to allow access to the JTAG and trace via an external interface, ordinarily a USB connector. It used Intel® DCI technologies. There are two types of Intel® DCI hosting interfaces in the platform. Intel® DCI is implemented using two primary transport topologies:

- Intel® Direct Connect Interface Out of Band (DCI.OOB, formerly BSSB)
- Intel® DCI.USB2 and DCI.USB3 (formerly DbC)

In the beginning of Early-Boot Debug, an Open and Closed-Chassis Debug are required. Both JTAG (Open Chassis) and DCI.OOB (Closed Chassis) are available prior the first Platform Boot Stall. Intel® DCI.USB2 becomes available which enables before CSE Boot Stall. Then Intel® DCI.USB3 is available during in S0 power state.

Intel® System Studio is a Software tools suite for System and IoT Development. Further information on how to run Intel® DCI through Intel® System Studio System is available at https://software.intel.com/en-us/articles/system-debugging-via-direct-connect-interfacedci-of-intel-system-debug. It can be used to run Intel® DCI by using its component tool called Intel® System Debugger. More information about Intel® System Studio is available at https://software.intel.com/en-us/system-studio.

NOTE

Intel® DCI and USB 3.1 Gen2 based debugger (kernel level debugger) are mutually exclusive.

In summary, Intel® DCI supports capabilities as below:

- Closed Chassis Debug at S0 and Sx State
- JTAG Access and Run-control (Probe Mode)
- System Tracing with Intel® Trace Hub

Debug host software that support Intel® DCI is:

- Intel® System Studio (ISS)
- TRACE32 by Lauterbach

Intel® DCI.OOB (Out of Band)

Intel® DCI.OOB was developed to provide an alternate path to convey controls and data to or from Intel® Trace Hub by connecting physically to the target through a USB 3.1 Gen2 port over Type A receptacle. Intel® DCI.OOB provides an alternate side band path around the USB 3.1 controller, so that the embedded logic can be accessed, even when the USB 3.1 controller is not alive (such as in low power states) or is malfunctioning. This path does not rely on USB 3.1 Gen2 protocol, link layer, or physical layer, because the xHCI functions are generally not available in such conditions.

Instead, this path relies on a special adapter that was developed by Intel called Intel® SVT Closed Chassis Adapter (CCA). It is a simple data transformation device. This adapter works together with debug host software and the embedded logic, contain a back-pressure scheme that makes both sides tolerant of overflow and starvation conditions, which is equivalent of USB 3.1 Gen2 link layer. This path also use native Intel® DCI packet protocol instead of USB 3.1 Gen2 protocol.

Intel® SVT CCA (MM Number: 921521) can be purchased through Intel® Design-In Tools Store at https://designintools.intel.com/product_p/itpxdpsvt.htm.

Besides Intel® SVT CCA, Lauterbach is an example of Third Party Vendor (TPV) solution. User may use a specific Lauterbach hardware and software configuration to connect between the Debug Host System and the Target Platform. It need Debug Host System (Lauterbach CombiProbe) to be in Downstream Facing Port (DFP) mode for Intel® DCI.OOB support in S0ix and Debug Host System (Lauterbach CombiProbe) can be DFP or Upstream Facing Port (UFP) for Intel® DCI.OOB supports in S0. Intel® SVT CCA (MM#:921521) can be purchased through Intel® Design-In Tools Store at https://designintools.intel.com/product_p/itpxdpsvt.htm.

Intel® DCI.USB2 and DCI.USB3

Intel® DCI USB2 and DCI.USB3 is a USB hosted Intel® DCI transport and the higher USB bandwidths, multiple parallel pipes or endpoints and BULK-mode data-integrity and retry-recovery mechanisms built into the protocol.

Supported USB endpoints include:

- DFx for JTAG/Run-control IA cores in system
- General Purpose 1 (GP1) for Kernel Mode Debug (KMD)
- General Purpose 2 (GP2) for Direct Memory Access (DMA) to system memory
- Trace (TRC) for streaming of live tracker

Intel® DCI.USB2 and DCI.USB3 supports multiple USB transport modes.

- **Intel® DCI.USB2** - Provides limited ~35 MB/s usable bandwidth, but extends USB hosting to cover early-boot and low power Sx and S0ix states.
- **Intel® DCI.USB3** - Provides an increase in S0 bandwidth up to ~800 MB/s usable bandwidth (generally limited further by host and host SW).

2.5 References

Specification	Document Number/Location
IEEE Standard Test Access Port and Boundary Scan Architecture	http://standards.ieee.org/findstds/standard/1149.1-2013.html
IEEE Standard for Boundary-Scan Testing of Advanced Digital Networks	https://standards.ieee.org/standard/1149_6-2015.html

3.0 Power Management

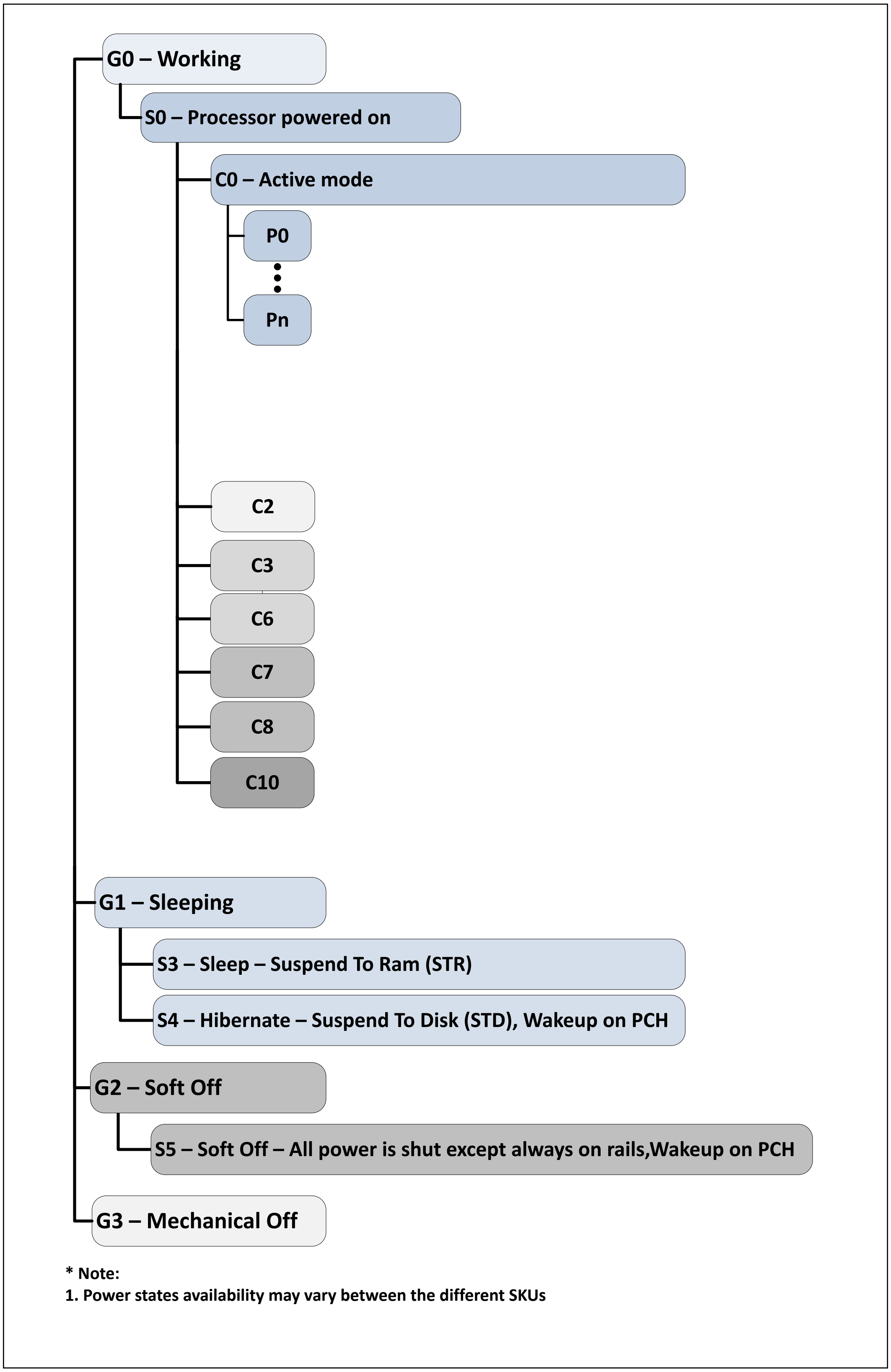
This chapter provides information on the following topics:

- Power Management States Supported
- Processor IA Core Power Management
- PM Interface Signals
- Processor Voltage Rails
- Voltage Rail Electrical Specifications

3.1 Power Management States Supported

This section describes the ACPI states supported by the processor.

Figure 2. System Power States


G0 – Working
S0 – Processor powered on
C0 – Active mode
P0
Pn
C2
C3
C6
C7
C8
C10
G1 – Sleeping
S3 – Sleep – Suspend To Ram (STR)
S4 – Hibernate – Suspend To Disk (STD), Wakeup on PCH
G2 – Soft Off
S5 – Soft Off – All power is shut except always on rails,Wakeup on PCH
G3 – Mechanical Off

*** Note:**

1. Power states availability may vary between the different SKUs

This figure shows how the platform ACPI states work with the CPU C power states (package C-states) and the CPU P performance states.

The following table describes the Gx/Sx ACPI states.

Table 2. System States

State	Description
G0/S0	Full On
G1/S3	Sleep/Suspend-to-RAM (STR). Context saved to memory
G1/S4	Suspend-to-Disk (STD). All power lost (except wake-up on PCH).
G2/S5	Soft off. All power lost (except wake-up on PCH). Total reboot.
G3	Mechanical off. All power removed from system.

The following table provides information on the IMC states.

Table 3. Integrated Memory Controller (IMC) States

State	Description
Power up	CKE asserted. Active mode
Pre-charge Power down	CKE de-asserted (not self-refresh) with all banks closed
Active Power down	CKE de-asserted (not self-refresh) with minimum one bank active
Self-Refresh	CKE de-asserted using device self-refresh

The following table provides information on how the Global and Sleep states relate to the Processor states and system clocks.

Table 4. G, S, and C Interface State Combinations

Global (G) State	Sleep (S) State	Processor Package (C) State	Processor State	System Clocks	Description
G0	S0	C0	Full On	On	Full On
G0	S0	C2	Deep Sleep	On	Deep Sleep
G0	S0	C3	Deep Sleep	On	Deep Sleep
G0	S0	C6/C7	Deep Power Down	On	Deep Power Down
G0	S0	C8	Off	On	Deeper Power Down
G0	S0ix	C10	Off	Off, except RTC	Enters S0ix
G1	S3	Power off	Off	Off, except RTC	Suspend to RAM
G1	S4	Power off	Off	Off, except RTC	Suspend to Disk
G2	S5	Power off	Off	Off, except RTC	Soft Off
G3	N/A	Power off	Off	Power off	Hard off

Table 5. State Transition Rules for the PCH

Present State	Transition Trigger	Next State
G0/S0/C0	• OPI Msg • SLP_EN bit set • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/Cx • G1/Sx or G2/S5/S4 state • G2/S5 • G3
G0/S0/Cx	• OPI Msg • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0 • S5 • G3
G1/S3	• Any Enabled Wake Event • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0[2] • G2/S5/S4 • G3
G1/S4	• Any Enabled Wake Event • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0[2] • G2/S5 • G3
G2/S5	• Any Enabled Wake Event • Mechanical Off/Power Failure	• G0/S0/C0[2] • 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 S3–S5 or G3 states to the S0 state are deferred until PMC_BATLOW_N is inactive in mobile configurations.
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.

The System has several independent power planes as described in the table. When a particular power plane is shut off, it should go to a 0 V level.

Table 6. System Power Plane

Plane	Controlled By	Description
CPU	PMC_SLP_S3_N signal	The PMC_SLP_S3_N signal is used to cut the power to the CPU completely.
Main (Applicable to Platform, PCH does not have a Main well)	PMC_SLP_S3_N signal	When PMC_SLP_S3_N goes active, power can be shut off to any circuit not required to wake the system from the S3 state. Since the S3 state requires that the memory context be preserved, power must be retained to the main memory. The processor, PCI Express* will typically be power-gated when the Main power plane is shut, although there may be small subsections powered. *Note:* The PCH power is not controlled by the PMC_SLP_S3_N signal, but instead by the PMC_SLP_SUS_N signal.
Device and Memory	PMC_SLP_S4_N signal PMC_SLP_S5_N signal	When PMC_SLP_S4_N 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 PMC_SLP_S5_N 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.

continued...

Plane	Controlled By	Description
Primary/ Suspend Well	PMC_SLP_SUS_N	This signal is asserted when the Primary/Suspend rails can be externally shut off for enhanced power saving
VCCIO_EXT	CPU_C10_GATE_N	This signal is asserted (LOW) when the processor enters C10 and can handle VCCIO_EXT,VCC1P8A, and VCCPLL_OC being lowered to 0V.
DEVICE[n]	Implementation Specific	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.

3.2 Processor IA Core Power Management

While executing code, Enhanced Intel SpeedStep® Technology and Intel® Speed Shift technology optimizes the processor’s IA core frequency and voltage based on workload. Each frequency and voltage operating point is defined by ACPI as a P-state. When the processor is not executing code, it is idle. A low-power idle state is defined by ACPI as a C-state. In general, deeper power C-states have longer entry and exit latencies but higher power savings.

NOTE

The performance configuration requires special tuning or adjustment of specific power management features.

3.2.1 OS/HW Controlled P-states

Enhanced Intel SpeedStep® Technology

Enhanced Intel SpeedStep® Technology enables OS to control and select P-state. For more information, refer to Enhanced Intel SpeedStep® Technology on page 26.

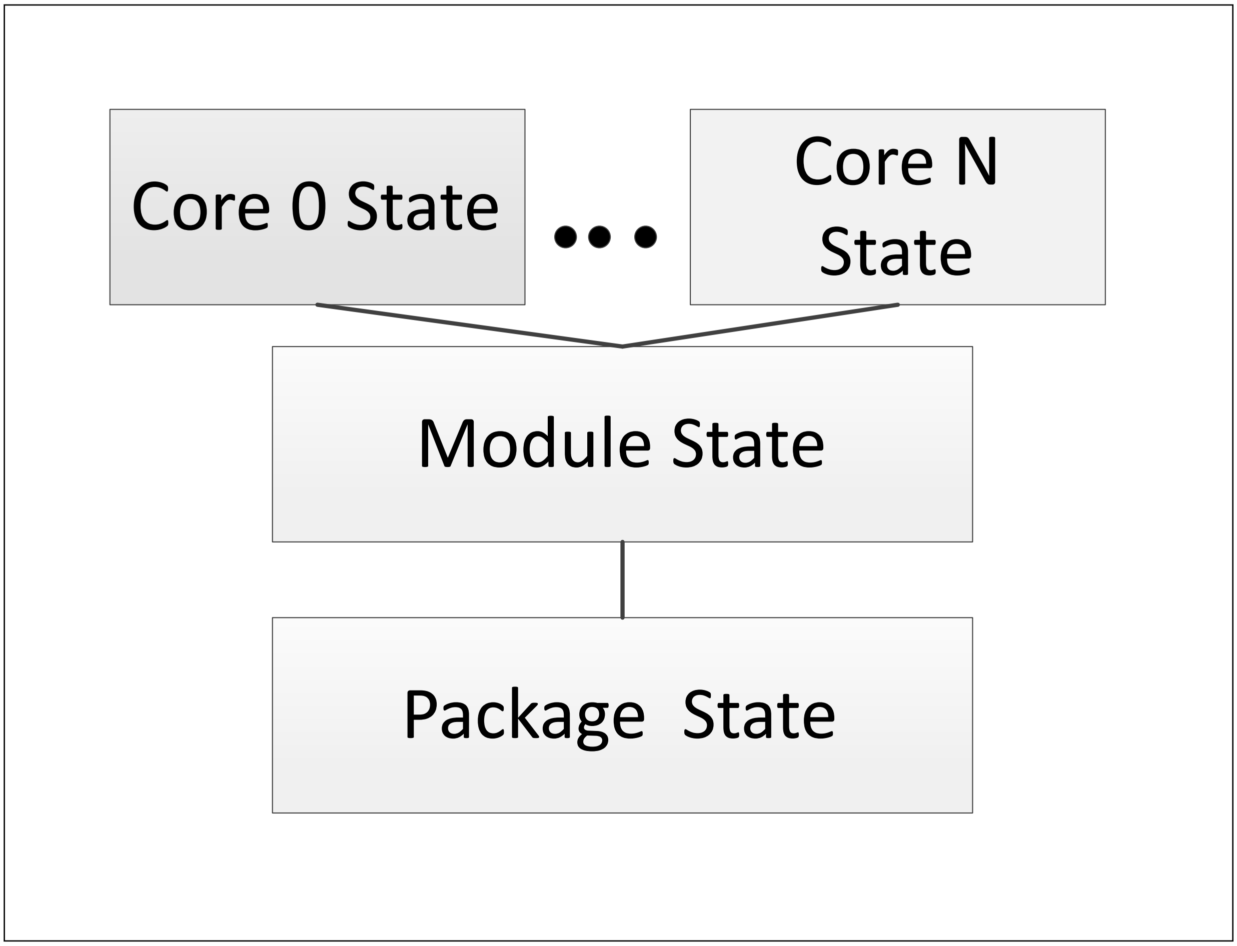
Intel® Speed Shift Technology

Intel® Speed Shift Technology is an energy efficient method of frequency control by the hardware rather than relying on OS control. Intel® Speed Shift Technology on page 26.

3.2.2 Low-Power Idle States

When the processor is idle, low-power idle states (C-states) are used to save power. More power savings actions are taken for numerically higher C-states (deeper C-states). However, deeper C-states have longer exit and entry latencies. Resolution of C-states occur at the thread, processor IA core, and processor package level.

Figure 3. Idle Power Management Breakdown of the Processor IA Cores



Processor IA core C-states are automatically resolved by the processor. A transition to and from C0 state is required before entering any other C-state.

3.2.3 Requesting Low-Power Idle States

The primary software interfaces for requesting low-power idle states are through the MWAIT instruction with sub-state hints and the HLT instruction (for C1 and C1E). However, software may make C-state requests using the legacy method of I/O reads from the ACPI-defined processor clock control registers, referred to as P_LVLx. This method of requesting C-states provides legacy support for operating systems that initiate C-state transitions using I/O reads.

For legacy operating systems, P_LVLx I/O reads are converted within the processor to the equivalent MWAIT C-state request. Therefore, P_LVLx reads do not directly result in I/O reads to the system. The feature, known as I/O MWAIT redirection, should be enabled in the BIOS.

The BIOS can write to the C-state range field of the PMG_IO_CAPTURE MSR to restrict the range of I/O addresses that are trapped and emulate MWAIT like functionality. Any P_LVLx reads outside of this range do not cause an I/O redirection to MWAIT(Cx) like request. They fall through like a normal I/O instruction.

When P_LVLx I/O instructions are used, MWAIT sub-states cannot be defined. The MWAIT sub-state is always zero if I/O MWAIT redirection is used. By default, P_LVLx I/O redirections enable the MWAIT 'break on EFLAGS.IF' feature that triggers a wake up on an interrupt, even if interrupts are masked by EFLAGS.IF.

3.2.4 Processor IA Core C-State Rules

The following are general rules for all processor IA core C-states, unless specified otherwise:

- A processor IA core transitions to C0 state when:
  - An interrupt occurs
  - There is an access to the monitored address if the state was entered using an MWAIT/Timed MWAIT instruction.
  - The deadline corresponding to the Timed MWAIT instruction expires.
- Any interrupt coming into the processor package may wake any processor IA core.
- A system reset re-initializes all processor IA cores.

Table 7. Core C-States

Core C-State	C-State Request Instruction	Description
C0	N/A	The normal operating state of a processor IA core where code is being executed
C1	MWAIT(C1)	AutoHalt - core execution stopped, autonomous clock gating (package in C0 state)
C1E	MWAIT(C1E)	Core C1 + lowest frequency and voltage operating point (package in C0 state)
C6	MWAIT(C6)	C6: Halt execution, flush core caches, flush core state, stop clock distribution, turn core voltage off
C6S	MWAIT(C6S)	C6S: C6 + allow entry to MC6
C7-C8	MWAIT(C7/C8)	Same as C6S, shrink the LLC
C10	MWAIT(C10)	Same as C6S, LLC flushed. Enable S0ix

Core C-State Auto-Demotion

In general, deeper C-states, such as C6, have long latencies and have higher energy entry/exit costs. The resulting performance and energy penalties become significant when the entry/exit frequency of a deeper C-state is high. Therefore, incorrect or inefficient usage of deeper C-states have a negative impact on battery life and idle power. To increase residency and improve battery life and idle power in deeper C-states, the processor supports C-state auto-demotion.

C-State auto-demotion:

- C6 to C1/C1E

The decision to demote a processor IA core from C6 to C1/C1E is based on each processor IA core's immediate residency history. Upon each processor IA core C6 request, the processor IA core C-state is demoted to C1 until a sufficient amount of residency has been established. At that point, a processor IA core is allowed to go into

C6. If the interrupt rate experienced on a processor IA core is high and the processor IA core is rarely in a deep C-state between such interrupts, the processor IA core can be demoted to a C1 state.

This feature is disabled by default. Refer the Firmware Architecture Specifications for more details on how to enable this.

There are also Module C-states related to the core C states.

Table 8. Module C-States

Module C-State	Description
MC0	At least one core in C0
MC6	All cores in C6 (powered off) CPLL bypassed (powered off) L2 flushed, L2 voltage = powered off

3.2.5 Package C-States

The processor supports C0, C2, C3, C6, C7, C8, and C10 package states.

The following is a summary of the general rules for package C-state entry. These apply to all package C-states, unless specified otherwise:

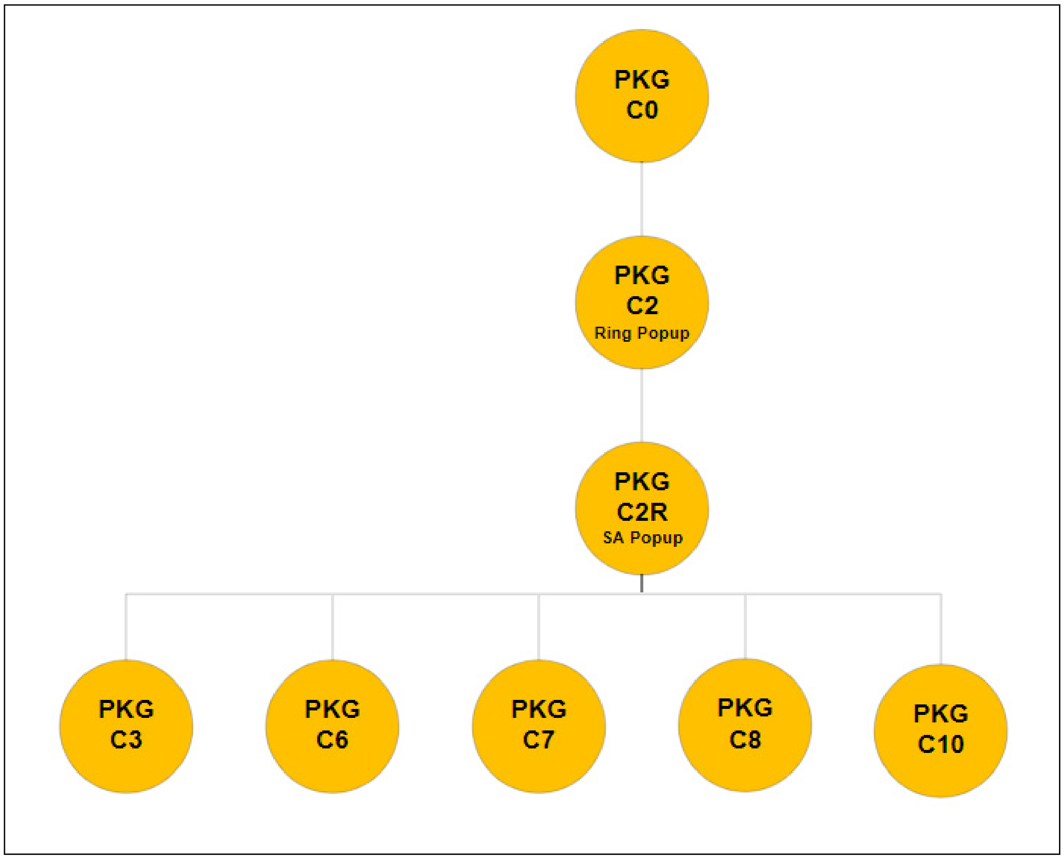
- A package C-state request is determined by the lowest numerical processor IA core C-state amongst all processor IA cores and also the module C-state.
- A package C-state is automatically resolved by the processor depending on the processor IA core idle power states and the status of the platform components.
  - Each processor IA core can be at a lower idle power state than the package if the platform does not grant the processor permission to enter a requested package C-state.
  - The platform may allow additional power savings to be realized in the processor.
  - For package C-states, the processor is not required to enter C0 before entering any other C-state.
  - Entry into a package C-state may be subject to auto-demotion – that is, the processor may keep the package in a deeper package C-state then requested by the operating system if the processor determines, using heuristics, that the deeper C-state results in better power/performance.

The processor exits a package C-state when a break event is detected. Depending on the type of break event, the processor does the following:

- If a processor IA core break event is received, the target processor IA core is activated and the break event message is forwarded to the target processor IA core.
  - If the break event is not masked, the target processor IA core enters the processor IA core C0 state and the processor enters package C0.
  - If the break event is masked, the processor attempts to re-enter its previous package state.
- If the break event was due to a memory access or snoop request,

- — But the platform did not request to keep the processor in a higher package C-state, the package returns to its previous C-state.
- — And the platform requests a higher power C-state, the memory access or snoop request is serviced and the package remains in the higher power C-state.

Figure 4. Package C-State Entry and Exit



The package level C states C3 through C10 are entered and exited through the C2R state.

Table 9. Package C-States

Package C State	Description
C0	Processor active state
C2	Cannot be requested explicitly by the software. All processor IA cores in C6 or deeper + Processor Graphic cores in RC6, memory path may be open. The processor will enter Package C2 when: • Transitioning from Package C0 to deep Package C state or from deep Package C state to Package C0. • All IA cores requested C6 or deeper + Processor Graphic cores in RC6 but there are constraints (LTR, programmed timer events in the near future and so forth) that prevent entry to any state deeper than C2 state.

continued...

Package C State	Description
	• All IA cores requested C6 or deeper + Processor Graphic cores in RC6 but a device memory access request is received. Upon completion of all outstanding memory requests, the processor transitions back into a deeper package C-state.
C2R	A transitional package C-State
C3	All cores in C6 or deeper + Processor Graphics in RC6, LLC may be flushed and turned off, memory in self refresh, memory clock stopped. The processor will enter Package C3 when: • All IA cores in C6 or deeper + Processor Graphic cores in RC6. • The platform components/devices allows proper LTR for entering Package C3.
C6	Package C3 + BCLK is off + IMVP VRs voltage reduction/PSx state is possible. The processor will enter Package C6 when: • All IA cores in C6 or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C6.
C7	Package C6 + If all IA cores requested C7, LLC ways may be flushed until it is cleared. If the entire LLC is flushed, voltage will be removed from the LLC. The processor will enter Package C7 when: • All IA cores in C7 or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C7.
C7S	Package C6 + If all IA cores requested C7S, LLC is flushed in a single step, voltage will be removed from the LLC. The processor will enter Package C7S when: • All IA cores in C7S or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C7S.
C8	Package C7 + LLC should be flushed at once. The processor will enter Package C8 when: • All IA cores in C8 or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C8.
C10	All processor die VR's are in lowest PS state or LPM + 38.4 MHz clock off. The processor will enter Package C10 when: • All IA cores in C10 + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C10.

Package C-State Auto-Demotion

The Processor may demote the Package C-state(s) to a shallower C-state(s), for example instead of going into package C10, it will demote to package C8 (and so on as required). The processor decision to demote the package C-state is based on the required C-states latencies, entry/exit energy/power and devices LTR.

Relevant S0ix

Processor supports the following S0ix variants. As the system goes deeper into S0ix, the overall functionality reduces, thereby, reducing the total power consumption. Longer S0ix residency gives better battery performance for a mobile/hand-held device.

Modern Standby is a relevant platform state in Windows*. Other relevant S0ix states exist on other OS. On display time out, the OS requests the processor to enter the package C10 state and platform devices at RTD3 (or disabled) in order to attain low power in idle. Relevant S0ix states require proper BIOS and OS configuration.

Table 10. S0ix Power States

Power State	Description	CPU Package State	Power Action	System Power States
LP1	Fully running S0 with aggressive opportunistic power management actions	C0	OPI L1 and PLL shutdown Individual PLL shutdown [1] Internal power gating of PCH controllers [2] Internal HSIO per lane power gating	S0
LP2	Pervasively Idle S0 and Root PLLs are off	C6 or deeper	All actions from LP1 + Gen 2 PLL/BCLK PLL shutdown	S0i2
LP3	Idle Floor	C10	All actions from LP2 + XTAL shutdown SLP_S0# VCCPRIM_CORE Low Voltage Mode	S0i3

Notes:
- Individual PLL shutdown - Each I/O interface when becoming sufficiently idle (typically requiring a minimum link power state) can have its respective I/O PLL be shutdown dynamically. This includes PCIe Gen3, SATA, USB 2.0 and MIPI.
- Internal Power Gating of PCH controllers - Each host controller (that is, xHCI, AHCI), PCIe* root port or embedded subsystem (Intel® CSE, Audio) when becoming sufficiently idle can autonomously power gate its core digital logic and local memory arrays. xHCI power gating is on a per port basis.

3.2.6 Exiting Sleep States

Sleep states (S3–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 hard disk 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 PCH-controlled Sleep states, the WAK_STS bit is set. The possible causes of wake events (and their restrictions) are shown in table below.

NOTE

(Mobile Only) If the BATLOW# signal is asserted, the PCH does not attempt to wake from an S3–S5 state, nor will it exit from Deep Sx 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 PCH, and the system wakes after BATLOW# is de-asserted.

Table 11. Cause of Wake Events

Cause	How Enabled	Wake from Sx	Wake from Deep 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	Yes	No
Power Button	Always enabled as Wake event.	Yes	Yes	Yes	Yes

continued...

Cause	How Enabled	Wake from Sx	Wake from Deep Sx	Wake from Sx After Power Loss[2]	Wake from "Reset" Types[3]
Any GPIOs except DSW GPIOs can be enabled for wake	Refer Note 5	Yes	No	No	No
LAN_WAKE_N [8]	Enabled natively (unless pin is configured to be in GPIO mode)	N/A	N/A	N/A	N/A
Intel ® High Definition Audio	Event sets PME_B0_STS bit; PM_B0_EN must be enabled. Can not wake from S5 state if it was entered due to power failure or power button override.	Yes	No	Yes	No
Primary PME#	PME_B0_EN bit in GPE0_EN[127:96] register.	Yes	No	Yes	No
Secondary PME#	Set PME_EN bit in GPE0_EN[127:96] register.	Yes	No	Yes	No
PCI Express* WAKE# pin	PCIEXP_WAKE_DIS bit.	Yes	Yes	Yes	No
SMBALERT#	Refer Note 4	Yes	No	Yes	Yes
SMBus Slave Wake Message (01h)	Wake/SMI# command always enabled as a Wake event. *Note:* SMBus Slave Message can wake the system from S3-S5, as well as from S5 due to Power Button Override.	Yes	No	Yes	Yes
SMBus Host Notify message received	HOST_NOTIFY_WKEN bit SMBus Slave Command register. Reported in the SMB_WAK_STS bit in the GPE0_STS register.	Yes	No	Yes	Yes
Wake Alarm Device	WADT_EN in GPE0_EN[127:96]	Yes	Yes	No	No

continued...

Cause	How Enabled	Wake from Sx	Wake from Deep Sx	Wake from Sx After Power Loss[2]	Wake from "Reset" Types[3]
AC_PRESENT	AC_PRESENT_WAKE_EN (Refer Note 6)	No	Yes	No	No
USB connection in/after deep-Sx	GPE0_EN.USB_CON_DSX_EN+	Refer Note 7	Yes	No	No

Notes:

1. If BATLOW# signal is low, PCH will not attempt to wake from S3-S5 (nor will it exit Deep Sx), even if valid wake event occurs. This prevents the system from waking when battery power is insufficient to wake the system. However, once BATLOW# goes back high, the system will boot.
2. This column represents what the PCH 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® CSE-initiated power button override, Intel® CSE-initiated host partition reset with power down, Intel® CSE Watchdog Timer, SMBus unconditional power down, processor thermal trip, PCH 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 or Deep Sx entry 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 master 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.
6. A change in AC_PRESENT causes an exit from Deep Sx to Sx, but the system will not wake all the way to S0.
7. Connection of a USB device can cause a wake from normal Sx as well. But that class of wakes is routed through PME_B0, not through this wake enable. The USB_CON_DSX_EN applies only to connection wakes while in Deep Sx or while in Sx after Deep Sx.

   Sx after Deep Sx reached due to AC_PRESENT going high while in Deep Sx, if Deep Sx is only enabled while on DC power.

   The following additional conditions are required for this wake to occur:

   a. The bit(s) in PM_CFG2.USB_DSX_PER_PORT_EN associated with the port(s) which experienced the connection must be set to '1'.

   b. DSX_CFG.USB_CON_DSX_MODE must be set to '1', routing USB connection to generate a wake rather than be reflected out to a pin.
8. No Integrated GbE

3.3 PM Interface Signals

The following table provides the list of power control signals used by the package.

Table 12. Signal Descriptions

Name	Type	Description
PMC_ACPRESENT	I	**AC Present:** Used on mobile systems to determine presence of AC power or battery power.
PMC_BATLOW_N	I	**Battery Low:** An input from the battery to indicate that there is insufficient power to boot the system. Assertion will prevent wake from S3–S5 state. This signal can also be enabled to cause an SMI# when asserted. This signal must be tied high to the VCCDSW_3p3, which will be tied to VCC_3P3A on this platform. *Note:* Require external Pull-up to VCCDSW_3p3.

continued...

Name	Type	Description
PMC_CORE_VID0	O	**PCH Core VID Bit 0:** May connect to discrete VR on platform and used to control the VCCIN_Aux rail (FIVR input) voltage. **In default mode this pin is driven high ('1')**
PMC_CORE_VID1	O	**PCH Core VID Bit 1:** May connect to discrete VR on platform and used to control the VCCIN_Aux rail (FIVR input) voltage. **In default mode this pin is driven high ('1')**
CPU_C10_GATE_N	O	External Power Gate control for VCCIO_EXT and VCCPLL_OC during C10. When asserted, VCCIO_EXT can be 0V, however the power good indicators for these rails must remain asserted.
PMC_DRAM_RESET_N	O	**System Memory DRAM Reset:** Active low reset signal, controls reset to the memory subsystems (DDR4/LPDDR4). *Note:* An external Pull-up to the DRAM power plane is required.
PMC_DSW_PWROK	IO	**DSW PWROK:** Power OK Indication for the VCC_3P3A_DSW voltage rail. This signal must be asserted no earlier than 10ms after the DSW power wells are valid.
PMC_PCH_PWROK	IO	**PCH Power OK:** When asserted, PMC_PCH_PWROK is an indication to the PCH that all of its core power rails have been stable for at least 5 ms. PMC_PCH_PWROK can be driven asynchronously. When PMC_PCH_PWROK is negated, the PCH asserts PMC_PLTRST_N. *Note:* PMC_PCH_PWROK must not glitch, even if PMC_RSMRST_N is low.
PMC_PLTRST_N	O	**Platform Reset:** The PCH asserts PMC_PLTRST_N to reset devices on the platform (such as SIO, LAN, processor, and so forth.). The PCH asserts PMC_PLTRST_N low in Sx states and when a cold, warm, or global reset occurs. The PCH de-asserts PMC_PLTRST_N upon exit from Sx states and the aforementioned resets. There is no guaranteed minimum assertion time for PMC_PLTRST_N. *Note:* PCI/PCIe* specification requires that the power rails associated with PCI/PCIe* (typically the 3.3 V, 5 V, and 12 V core well rails) have been valid for 100 ms prior to PMC_PLTRST_N de-assertion. System designers must ensure the requirement is met on the platform.
PMC_PWRBTN_N	I	**Power Button:** Power button input signal. Used to wake the processor from power button press. The Power Button will cause 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 PMC_PWRBTN_N 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 S3-S4 states. This signal has an internal Pull-up resistor and has an internal 16 ms de-bounce on the input.
PMC_RSMRST_N	I	**Resume Well Reset:** This signal is used for resetting the resume power plane logic. This signal must be asserted for at least 10 ms after the suspend power wells are valid. When de-asserted, this signal is an indication that the suspend power wells are stable.
PMC_SLP_WLAN_N	O	**WLAN Sub-System Sleep Control:** When PMC_SLP_WLAN_N is asserted, power can be shut off to the external wireless LAN device. PMC_SLP_WLAN_N will always will be de-asserted in S0.
PMC_SLP_S0_N	O	**S0 Sleep Control:** When PCH is idle and 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 an external power management controller for other power management related optimizations.
PMC_SLP_S3_N	O	**S3 Sleep Control:** PMC_SLP_S3_N is for power plane control. This signal shuts off power to all non-critical systems when in S3 (Suspend To RAM), S4 (Suspend to Disk), or S5 (Soft Off) states.

continued...

Name	Type	Description
PMC_SLP_S4_N	O	**S4 Sleep Control:** PMC_SLP_S4_N is for power plane control. This signal shuts power to all non-critical systems when in the S4 (Suspend to Disk) or S5 (Soft Off) state.
PMC_SLP_S5_N	O	**S5 Sleep Control:** PMC_SLP_S5_N is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) states.
PMC_SUSCLK	O	**Suspend Clock:** This clock is a digitally buffered version of the RTC clock.
PMC_SUSPWRDNACK	O	**SUSPWRDNACK:** Active high. Asserted by the PCH when it does not require the PCH Primary well to be powered.
PMC_SYS_PWROK	I	**System Power OK:** This generic power good input to the PCH is driven and utilized in a platform-specific manner. While PMC_PCH_PWROK always indicates that the core wells of the PCH are stable, PMC_SYS_PWROK is used to inform the PCH that power is stable to other required system component(s) and the system is ready to start the exit from reset. (de-asserts PMC_PLTRST_N to the processor).
PMC_SYS_RESET_N	I	**System Reset:** Reset button input signal to reset the processor. This pin forces an internal reset after being debounced. *Note:* External pull-up resistor required
PMC_VRALERT_N	I	**VR Alert:** ICC Max. throttling indicator from the PCH voltage regulators. PMC_VRALERT_N pin allows the VR to force throttling to prevent an over current shutdown.
PMC_WAKE_N	I/O	**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 Output pin during S0IX states hence this pin can not be used to wake up the system during S0IX states. • External Pull-up required.
PMC_ALERT_N	I	PD controller's USB-C interrupt request is presented to PMC as processor USB-C Mux Manager, through PMC_ALERT_N pin assertion. *Note:* External pull-up resistor required
VCCST_OVERRIDE	O	This signal is used in Debug and Class. No usage in any of the functional scenarios.
THRMTRIP_N	O	**Thermal Trip:** Asserted during a catastrophic thermal event. Platform design should restart or shut down the voltage rails after this event. For platform using discrete VR (voltage regulator) power delivery solution, there is additional platform logic required to initiate VR shut down on the platform when this signal is asserted.
PCHHOT_N	OD	PCHHOT_N indicates that it has exceeded some temperature limit set by BIOS. The temperature limit (programmed into the PHL register) is compared to the present temperature. If the present temperature is greater than the PHL value then the pin is asserted.

3.4 Processor Voltage Rails

This section contains information about the following:

- Fully Integrated Voltage Regulator (FIVR)
- Main Platform Voltage Regulators
- Additional Voltage Rail Signals
- VCCIN_AUX
- External Bypass Rails (VCC_VNNEXT_1P05 and VCC_V1P05EXT_1P05)

3.4.1 Fully Integrated Voltage Regulator (FIVR)

The processor integrates multiple voltage rails in order to reduce BOM costs for the platform, and to enable additional voltage level features the processor can take advantage of.

There are FIVRs integrated on the PCH, VNN/V1P05 which is sourced from VCCIN_Aux. VCCIN_Aux also sourced the VCCSA rail in a CPU. In addition to VCCSA FIVR, compute die integrates 4 additional FIVRs to source VCCCORE, VCCSA, VCCL2, VCCGT and VCCRING, which derives the respective voltages from VCCIN VR on platform. Each FIVR is able to control a specific voltage rail.

3.4.2 Main Platform Voltage Regulators

In the table below are the main platform voltage rails that are regulated and controlled on the platform.

Table 13. Platform Voltage Rails

Rail	Voltage	Description
VCCIN	0 V (MIN) and 2 V (MAX)	On-Package VR (OPVR) power rail
VCCIN_Aux	1.65 or 1.8 V - Active 1.1 V - Retention Off - Idle States	On-Package VR (OPVR) power auxiliary rail
VDDQ	1.2 V (DDR4) or 1.1 V (LPDDR4x)	System memory power rail
VCCPRIM_3P3	3.3 V	Primary 3.3 V power supply
VCCPRIM_1P8	1.8 V	Primary 1.8 V power supply
VCC_VNNEXT_1P05 (Optional)	1.05 V or 0.76 V	Used for FIVR PRIM_CORE bypass mode
VCC_V1P05EXT_1P05 (Optional)	1.05 V	Used for FIVR PCH IO bypass mode
VCCIO_EXT	1 V (Typ)	VCCIO_EXT rail

3.4.3 Additional Voltage Rail Signals

There are additional voltage rail pins for routing power between parts of the processor and the platform listed in the table below.

Table 14. Additional Voltage Rail Signals

Rail	Voltage	Description
VCCPLL	1.05 V	Processor PLL power rails
VCCST	1.05 V	Sustain voltage for processor standby modes
VCC1P05_OUT	1.05 V	FIVR Output of PCH to platform 1.05 V Power Gates
VCCSTG	1.05 V	Gated sustain voltage for processor standby modes
VCCSTG_OUT	1.05 V	VCCSTG_OUT Power rail

continued...

Rail	Voltage	Description
VCCLDOSTD_0P85	0.85 V	This rail is generated internally and needs to be routed out to the motherboard for decoupling purpose.
VCCPRIM_1P05	1.05 V	1.05 V Primary rail
VCCA_CLKLDO_1P8	1.8 V	Analog supply for internal clocks
VCCDPHY_1P24	1.24 V	1.24 V rail for DPHY
VCCPGPPR	1.8 V / 3.3 V	VCC_PGPPR is sourced from either VCC_3P3A or VCC1P8A rail and this rail is input supply for the GP_R GPIOs.
VCC1P8A	1.8 V	1.8A rail. This rail is turned off when either PMC_CPU_C10_GATE_N or PMC_SLP_S3_N is asserted (LOW). Thus, this rail is off in package C10 state as well as S3 - S5 states.
VCCRTC	3 V	RTC Well Supply. This rail can drop to 2.0 V if all other planes are off. This power is not expected to be shut off unless the RTC battery is removed or drained. *Note:* VCCRTC nominal voltage is 3.0 V. This rail is intended to always come up first and always stay on. It should NOT be power cycled regularly on non-coin battery designs. *Note:* Implementation should not attempt to clear CMOS by using a jumper to pull VCCRTC low. Clearing CMOS can be done by using a jumper on RTCRST# or GPI.
VCCPLL_OC	1.1 V/1.2 V	Processor PLLs power rails
VCCSPI	3.3 V or 1.8 V	SPI Primary Well 3.3 V or 1.8 V. If powered at 3.3 V, the 3.3 V supply can come from VCCPRIM_3P3 supply. If powered at 1.8 V, the 1.8 V supply can come from VCCPRIM_1P8 supply.
VCCDSW_3P3	3.3 V	3.3 V Deep Sx Well.

3.4.4 VCCIN_AUX

From the platform perspective, the FIVRs require an input rail to generate the internal voltage rails. This rail is referred to as VCCIN_AUX.For the PCH, the input regulator must be able to support at least 1.8 V. During the deep S0ix states, the input rail to the FIVRs can be disabled. This will be done by driving the CORE_VID values to ‘00. VCCIN_AUX powergood during initial reset is tied into the PMC_RSMRST_N signal, requiring that the FIVR input voltage rail is stable in the same window as the other PMC_SLP_SUS_N rails. Internal FIVRs will generate Vnn, V1P05 rails.

NOTE

Leakage from VCCIN_AUX is expected behavior when CORE_VID[1:0]=00; this leakage voltage may be as high as 1.15 V during Sx and S0ix states

3.4.5 External Bypass Rails (VCC_VNNEXT_1P05 and VCC_V1P05EXT_1P05)

The V1p05 and Vnn rails can also have an input from a separate external voltage rail. These rails are always on and must come up after the V1p8A rail has been brought up. Note that there is no feedback that this rail is valid.

3.5 Voltage Rail Electrical Specifications

The processor DC specifications in this section are defined at the processor signal pins, unless noted otherwise. Icc_max specifications are estimates on the currunt consumption by the processor pins only. Other additional devices that consume current on the platform need to be considered separately.

- The *Voltage and Current Specifications* section lists the DC specifications for the processor and are valid only while meeting specifications for junction temperature, clock frequency, and input voltages. Read all notes associated with each parameter.
- AC tolerances for all DC rails include dynamic load currents at switching frequencies up to 1 MHz.

3.5.1 Processor Power Rails DC Specifications

Vcc_{IN} DC Specifications

Table 15. Processor Vcc_{IN} Active and Idle Mode DC Voltage and Current Specifications

Symbol	Parameter	Segment	Min	Typ	Max	Unit	Note[1]
Operating Voltage	Voltage Range for Processor Operating Mode	All	0	1.8	2	V	1,2,3, 7,11
IccMAX	Maximum Processor ICC	4 Core(6 W)	—	—	33	A	4,6,7,10
		2 Core(6 W)			22	A	
		4 Core(10 W)			35	A	
		2 Core(10 W)			25	A	
Icc_{TDP}	Thermal Design Current for processor VccIN Rail	—	—	—	10	A	
TOB_{VCC}	Voltage Tolerance	PS0,PS1	—	—	±20	mV	3, 6, 8
		PS2,PS3			±35		
Ripple	Ripple Tolerance	PS0,PS1			±15	mV	3, 6, 8
		PS2,PS3			±30		
DC_LL	Loadline slope within the VR regulation loop capability	—	0	—	2	mΩ	9,12,13
AC_LL	AC Loadline (<10 MHz)	—	—	—	5	mΩ	9,12,13
VOS	Max Overshoot Voltage				200	mV	
T_OVS_TDP_MAX	Max Overshoot time	—	—	—	500	µs	

continued...

Symbol	Parameter	Segment	Min	Typ	Max	Unit	Note[1]
	TDP/virus mode						
V_OVS TDP_MAX/ virus_MAX	Max Overshoot at TDP/virus mode	—	—	—	10	%	

Notes:
1. All specifications in this table are based on estimates and simulations or empirical data. These specifications will be updated with characterized data from silicon measurements at a later date.
2. Each processor is programmed with a maximum valid voltage identification value (VID) that is set at manufacturing and cannot be altered. Individual maximum VID values are calibrated during manufacturing such that two processors at the same frequency may have different settings within the VID range. Note that this differs from the VID employed by the processor during a power management event (Adaptive Thermal Monitor, Enhanced Intel SpeedStep Technology, or low-power states).
3. The voltage specification requirements are measured across Vcc_SENSE and Vss_SENSE as near as possible to the processor with an oscilloscope set to 100 MHz bandwidth, 1.5 pF maximum probe capacitance, and 1 MΩ minimum impedance. The maximum length of ground wire on the probe should be less than 5 mm. Ensure external noise from the system is not coupled into the oscilloscope probe.
4. Processor VccIN VR to be designed to electrically support this current.
5. Processor VccIN VR to be designed to thermally support this current indefinitely.
6. Long term reliability cannot be assured if tolerance, ripple, and core noise parameters are violated.
7. Long term reliability cannot be assured in conditions above or below Max/Min functional limits.
8. PSx refers to the voltage regulator power state as set by the SVID protocol.
9. LL measured at sense points.
10. Typ column represents IccMAX for commercial application it is NOT a specification - it is a characterization of limited samples using limited set of benchmarks that can be exceeded.
11. Operating voltage range in steady state.
12. LL spec values should not be exceeded. If exceeded, power, performance and reliability penalty are expected.
13. Load Line (AC/DC) should be measured by the VRTT tool and programmed accordingly via the BIOS Load Line override setup options. AC/DC Load Line BIOS programming directly affects operating voltages (AC) and power measurements (DC). A superior board design with a shallower AC Load Line can improve on power, performance and thermals.

VCC1P8A DC Specifications

Table 16. Processor VCC1P8A Supply DC Voltage and Current Specifications

Symbol	Parameter	Min	Typ	Max	Unit	Notes[1,2]
Vcc_{1p8A}	Package voltage (DC + AC specification)	—	1.8	—	V	1,3
Icc_{MAX_1p8A}	Max Current for Vcc_{1p8A} Rail	—	—	0.7	A	1
Icc_{idle}	Sx Icc Idle Current	—	—	100	mA	
TOB Vcc_{1p8A}	Vcc_{1p8A} Tolerance	AC+DC=+/-5%			%	1,3

Notes:
1. All specifications in this table are based on estimates and simulations or empirical data. These specifications will be updated with characterized data from silicon measurements at a later date.
2. Long term reliability cannot be assured in conditions above or below Max/Min functional limits.
3. The voltage specification requirements are measured on package pins as near as possible to the processor with an oscilloscope set to 100 MHz bandwidth, 1.5 pF maximum probe capacitance, and 1 MΩ minimum impedance. The maximum length of ground wire on the probe should be less than 5 mm. Ensure external noise from the system is not coupled into the oscilloscope probe.

VccIN_AUX DC Specifications

Table 17. VccIN_AUX Supply DC Voltage and Current Specifications

Symbol	Parameter	Min	Typ	Max	Unit	Note[1]
Vcc_{in_AUX}		0	1.8		V	1,3,4
Icc_{MAX}	Maximum VccIN_AUX Icc	0	—	24	A	1,10
Icc_{idle}	Sx Icc Idle Current	0	—	201	mA	
Icc_{TDP}	Thermal Design Current for processor VccIN_Aux Rail	—	—	4	A	
TOB_{VCC}	Voltage Tolerance Budget	—	—	AC+DC: -10/+5	%	1,3,6
VOS	Overshoot Voltage	—	—	1.95	V	7
TVOS	Overshoot Time	—	—	500	us	7

Notes:
1. All specifications in this table are based on estimates and simulations or empirical data. These specifications will be updated with characterized data from silicon measurements at a later date.
2. Long term reliability cannot be assured in conditions above or below Max/Min functional limits.
3. The voltage specification requirements are measured on package pins as near as possible to the processor with an oscilloscope set to 100 MHz bandwidth, 1.5 pF maximum probe capacitance, and 1 MΩ minimum impedance. The maximum length of ground wire on the probe should be less than 5 mm. Ensure external noise from the system is not coupled into the oscilloscope probe.
4. Max impedance allowed between 1 MHz-40 MHz is lower than LL3. Comply with recommended impedance target to avoid coupling noise concerns
5. The LL3 values are for reference. must still meet voltage tolerance specification.
6. Voltage Tolerance budget values Includes ripples
7. Overshoot with max voltage of 2.13 V is allowed if it sustained for less then 500 us.
8. This rail can be connect to 1.65 V
9. The ICCMAX values combine power pins that feed the compute die and the PCH die in the processor.

V_{DDQ} DC Specifications

Table 18. Memory Controller (V_{DDQ}) Supply DC Voltage and Current Specifications

Symbol	Parameter	Min	Typ	Max	Unit	Note[1]
$V_{DDQ\ (LPDDR4/x)}$	Processor I/O supply voltage for LPDDR4/x	-	1.1 (+/-5%)	-	V	3,4,5
$V_{DDQ\ (DDR4)}$	Processor I/O supply voltage for DDR4	-	1.2 (+/-5%)	-	V	3,4,5
TOB_{VDDQ}	VDDQ Tolerance	AC+DC:± 5%			%	3,4
Icc_{MAX_VDDQ} (LPDDR4/x)	Max Current for V_{DDQ} Rail (LPDDR4/x)	—	—	3.5	A	2
IccMAX_VDDQ (DDR4)	Max Current for VDDQ Rail	—	—	3.5	A	

continued...

Symbol	Parameter	Min	Typ	Max	Unit	Note[1]
	(DDR4)					

Notes: 1. All specifications in this table are based on estimates and simulations or empirical data. These specifications will be updated with characterized data from silicon measurements at a later date.
2. The current supplied to the DRAM is not included in this specification.
3. Includes AC and DC error, where the AC noise is bandwidth limited to under 100 MHz, measured on package pins.
4. No requirement on the breakdown of AC versus DC noise.
5. The voltage specification requirements are measured on package pins as near as possible to the processor with an oscilloscope set to 100 MHz bandwidth, 1.5 pF maximum probe capacitance, and 1 MO minimum impedance. The maximum length of ground wire on the probe should be less than 5 mm. Ensure external noise from the system is not coupled into the oscilloscope probe.

Additional Rails DC Characteristics

Table 19. Additional Rails Estimated I_{CC}

Voltage Rail	V Min	V Typical (V)	V Max	Iccmax Current[1] (A)	Sx Iccmax Current (mA)	Sx Icc Idle Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (uA)
VCCIO_EXT[3]	0.98	1	1.02	5	-	-	-	-
VCCST	0.97	1.05	1.07	0.6	-	-	-	-
VCCSTG	0.97	1.05	1.07	0.15	-	-	-	-
VCCPLL	0.97	1.05	1.07	0.1	-	-	-	-
VCCPLL_OC	-	1.1/1.2	-	0.1	-	-	-	-
VCC_VNNEXT_1P05	-	1.05	-	0.2	150	0	-	-
VCC_V1P05EXT_1P05	-	1.05	-	0.2	150	0	-	-
VCCPGPPR	-	1.8	-	0.000237	0.17437	0	-	-
VCCA_CLKLDO_1P8	-	1.8	-	0.12	7.83135	2.965083	-	-
VCCPRIM_1P8	-	1.8	-	1.3	619.379	124.8817	-	-
VCCPRIM_3P3	-	3.3	-	0.232	0.92362	0.260667	-	-
VCCRTC	-	3.3	-	0.001	2	0.605318	0.339	10
VCCDSW_3P3	-	3.3	-	0.001	0.32315	0.293773	2	-

Notes: 1. Iccmax estimates assumes IPs are working at Vmax with 100% activity at 105°C.
2. The Iccmax value is a steady state current that can happen after respective power ok has asserted (or reset signal has de-asserted).
3. VCCIO_EXT voltage tolerance is +/-5% for (DC+AC) & 2% only for DC

4.0 Thermal Management

This chapter contains information about the following:

- Thermal and Power Specifications
- Processor Thermal Management
- PCH Thermal Management

4.1 Thermal and Power Specifications

For more information on target processor specifications, refer to Table 1 on page 20

NOTE

The TDP values are the average power dissipation in junction temperature operating condition limit, for the SKU Segment and Configuration, for which the processor is validated during manufacturing when executing an associated Intel-specified TDP workload.

TDP workload may consist of a combination of processor IA core intensive and graphics core intensive applications.

4.2 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 junction temperature (Tj_{MAX}) specification at the maximum Thermal Design Power (TDP).
- 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.

4.2.1 Thermal Considerations

The processor TDP is the maximum sustained power that should be used for design of the processor thermal solution. TDP is a power dissipation and junction temperature operating 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. TDP 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 a PCH on a single package. This may result in power distribution differences across the package and should be considered when designing the thermal solution.

Intel® Burst Technology 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® Burst Technology is enabled:

- Applications are expected to run closer to TDP 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 TDP 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 Burst concept, where peak burst frequency is achieved only when 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 are designed to less than thermal design guidance may experience thermal and performance issues.

Package Power Control

The package power control settings of PL1, PL2, PL3, PL4 and Tau allow the designer to configure Intel® Burst Technology 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 TDP 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 (RAPL) 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.
- **Burst Time Parameter (Tau):** An averaging constant used for PL1 Exponential Weighted Moving Average (EWMA) power calculation.

Implementation of Intel® Burst Technology only requires configuring PL1, PL1 Tau and PL2. PL3 and PL4 are disabled by default.

Burst Time Parameter (Tau)

Burst Time Parameter (Tau) is a mathematical parameter (units of seconds) that controls the burst algorithm. During a maximum power burst event, the processor could sustain PL2 for a duration longer than the Burst Time Parameter. If the power value and/or Burst Time Parameter is changed during runtime, it may take some time based on the new Burst 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 Burst Time Parameter associated with Package Power Control and Platform Power Control.

4.2.2 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.

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 junction temperature Tj_{MAX}.

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.

Clock modulation (refer to Clock Modulation on page 57) is another means to reduce the processor core clock. The duty cycle of the clock modulation can be programmed through MSR (refer to MSR Based On-Demand Mode on page 61).

Tj_{MAX} is factory calibrated and is not user configurable. The default value is software visible in the TEMPERATURE_TARGET (0x1A2) 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 = TDP. The system design should provide a thermal solution that can maintain normal operation when PL1 = TDP within the intended usage range.

NOTE

Adaptive Thermal Monitor protection is always enabled.

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. 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 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 is not activated by the Package average temperature control mechanism.

Digital Thermal Sensor

Each processor has up to 12 on-die Digital Thermal Sensors (DTS) that detect the processor IA (with one sensor in the core), GT (nine sensors), CCU (one sensor), and display (one sensor).

Temperature values from the DTS can be retrieved through:

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

When temperature is retrieved by the processor MSR, it is the instantaneous temperature of the given DTS. 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 (0x1B1) MSR and IA32_THERM_STATUS (0x19C) MSR.

Code execution is halted in C1 or deeper C-states.

Unlike traditional thermal devices, the DTS outputs a temperature relative to the maximum supported operating temperature of the processor (Tj_{MAX}), 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 (0x1A2) MSR . The temperature returned by the DTS is an implied negative integer indicating the relative offset from Tj_{MAX}. The DTS does not report temperatures greater than Tj_{MAX}. 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 0x0, 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, one set above and another below the current temperature, located in the processor thermal MSRs. These thresholds have the capability of generating interrupts using the processor IA core's local APIC.

The thermal thresholds defined for Processor are:

- **Core Threshold #1 Temperature in IA32_THERM_INTERRUPT (MSR 0x19B) Bits 14:8**. This value indicates the offset in degrees below Tj_{MAX} Temperature that will trigger a Thermal Threshold 1 trip.
- **Package Threshold #1 Temperature in IA32_THERM_INTERRUPT (MSR 0x1B2) Bits 14:8**. This value indicates the offset in degrees below Tj_{MAX} Temperature that will trigger a Package Thermal Threshold 1 trip.
- **Core Threshold #2 Temperature in IA32_THERM_INTERRUPT (MSR 0x19B) Bits 22:16**. This value indicates the offset in degrees below Tj_{MAX} Temperature that will trigger a Thermal Threshold 2 trip. Similar to Threshold Value 1.

- **Package Threshold #2 Temperature in IA32_THERM_INTERRUPT (MSR 0x1B2) Bits 22:16**. This value indicates the offset in degrees below Tj_{MAX} Temperature that will trigger a Thermal Threshold 2 trip to all cores in the package. Similar to Core Threshold Value 2.

Digital Thermal Sensor Accuracy (Taccuracy)

The error associated with DTS measurements does not exceed ±5 °C within the entire operating range.

NOTE

DTS does not report negative values.

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. T_{FAN} temperature (sometimes called $T_{CONTROL}$) indicates the relative offset from the Thermal Monitor Trip Temperature at which fans should be engaged. For current temperature reporting, it is recommended that the value MSR PACKAGE_THERM_MARGIN (1A1h) [15:0] be used for fan control software. Intel recommends full cooling capability before the DTS reading reaches Tj_{MAX}.

PROCHOT_N Signal

PROCHOT_N (processor hot) is asserted by the processor when the TCC is active. Only a single PROCHOT_N pin exists at a package level. When any DTS temperature reaches the TCC activation temperature, the PROCHOT_N signal is asserted. PROCHOT_N assertion policies are independent of Adaptive Thermal Monitor enabling.

Bi-Directional PROCHOT_N

By default, the PROCHOT_N is configured as bi-directional pin. When configured as an input or bi-directional signal, PROCHOT_N is used for thermally protecting other platform components should they overheat as well. When PROCHOT_N is driven by an external device:

- The package will immediately transition to the lowest P-State (Pn) supported by the processor IA cores and graphics cores. This is contrary to the internally-generated Adaptive Thermal Monitor response.
- Clock modulation is not activated.

The processor package remains at the lowest supported P-state until the system de-asserts PROCHOT_N. The processor is configured to generate an interrupt upon assertion and de-assertion of the PROCHOT_N signal.

When PROCHOT_N is configured as a bi-directional signal and PROCHOT_N is asserted by the processor, it is impossible for the processor to detect a system assertion of PROCHOT_N. The system assertion will have to wait until the processor de-asserts PROCHOT_N before PROCHOT_N action can occur due to the system assertion. While the processor is hot and asserting PROCHOT_N, the power is reduced but the reduction rate is slower than the system PROCHOT_N response of < 100 us. The processor thermal control is staged in smaller increments over many milliseconds. This may cause several milliseconds of delay to a system assertion of PROCHOT_N while the output function is asserted.

Voltage Regulator Protection using PROCHOT_N

PROCHOT_N may be used for thermal protection of voltage regulators (VR). System designers can create a circuit to monitor the VR temperature and assert PROCHOT_N and, if enabled, activate the TCC when the temperature limit of the VR is reached. When PROCHOT_N is configured as a bi-directional or input only signal, if the system assertion of PROCHOT_N is recognized by the processor, it will result in an immediate transition to the lowest P-State (Pn) supported by the processor IA cores and graphics cores. Systems should still provide proper cooling for the VR and rely on bi-directional PROCHOT_N only as a backup 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 TDP.

Thermal Solution Design and PROCHOT_N Behavior

With a properly designed and characterized thermal solution, it is anticipated that PROCHOT_N 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 PROCHOT_N in the anticipated ambient environment may:

- Cause a noticeable performance loss.
- Result in prolonged operation at the specified maximum junction 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).

Low-Power States and PROCHOT_N Behavior

Depending on package power levels during package C-states, outbound PROCHOT_N 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 PROCHOT_N will re-assert, although typically package idle state residency should resolve any thermal issues.

THRMTRIP_N 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 THRMTRIP_N signal will go active.

Critical Temperature Detection

Critical Temperature detection is performed by monitoring the package temperature. This feature is intended for graceful shutdown before the THRMTRIP_N is activated. However, the processor execution is not guaranteed between critical temperature and THRMTRIP_N. 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 (0x1B1) MSR and the condition also generates a thermal interrupt, if enabled.

On-Demand Mode

The processor provides an auxiliary mechanism that allows system software to force the processor to reduce its power consumption using clock modulation. This mechanism is referred to as “On-Demand” mode and is distinct from Adaptive Thermal Monitor and bi-directional PROCHOT_N. The processor platforms should not rely on software usage of this mechanism to limit the processor temperature. On-Demand Mode can be accomplished using processor MSR or chipset I/O emulation. On-Demand Mode may be used in conjunction with the Adaptive Thermal Monitor. However, if the system software tries to enable On-Demand mode at the same time the TCC is engaged, the factory configured duty cycle of the TCC will override the duty cycle selected by the On-Demand mode. If the I/O based and MSR-based On-Demand modes are in conflict, the duty cycle selected by the I/O emulation-based On-Demand mode will take precedence over the MSR-based On-Demand Mode.

MSR Based On-Demand Mode

If Bit 4 of the IA32_CLOCK_MODULATION MSR is set to 1, the processor will immediately reduce its power consumption using modulation of the internal processor IA core clock, independent of the processor temperature. The duty cycle of the clock modulation is programmable using bits [3:0] of the same IA32_CLOCK_MODULATION MSR. In this mode, the duty cycle can be programmed in 6.25% increments. Thermal throttling using this method will modulate each processor IA core's clock independently.

I/O Emulation-Based On-Demand Mode

I/O emulation-based clock modulation provides legacy support for operating system software that initiates clock modulation through I/O writes to ACPI defined processor clock control registers on the chipset (PROC_CNT). Thermal throttling using this method will modulate all processor IA cores simultaneously.

4.2.3 Intel® Memory Thermal Management

The processor provides thermal protection for system memory by throttling memory traffic when using either DIMM modules or a memory down implementation. Two levels of throttling are supported by the processor, either a warm threshold or hot threshold that is customizable through memory mapped I/O registers:

- Throttling based on the warm threshold should be an intermediate level of throttling.
- Throttling based on the hot threshold should be the most severe.

The amount of throttling is dynamically controlled by the processor.

4.3 PCH Thermal Management

This section provides information on the following topics:

- PCH Thermal Sensor
- Modes of Operation
- Thermal Reporting to an External Device
- Temperature Trip Point
- Thermal Sensor Accuracy (Taccuracy

- Thermal Trip Signal (PCHHOT_N)

4.3.1 PCH Thermal Sensor

The PCH incorporates one on-die Digital Thermal Sensors (DTS) for thermal management.

4.3.2 Modes of Operation

The DTS has two usages when enabled:

1. Provide the PCH temperature in units of 1°C to the EC.
2. Allow programmed trip points to cause alerts via an interrupt (SCI and SMI) or shut down the system (unconditionally transitions the system to S5) with a programmable catastrophic trip point.

4.3.3 Thermal Reporting to an External Device

To support a platform EC that is managing the system thermals, the PCH provides the ability for an external device to read the PCH temperature over SMLink1 or over eSPI interface. The EC will issue an SMBus read or eSPI OOB Channel request and receives a single byte of data, indicating a temperature between 0°C and 127°C, where 255 (0xFF) indicates that the sensor is not enabled yet. The EC must be connected to SMLink1 for thermal reporting support.

4.3.4 Temperature Trip Point

The internal thermal sensor 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. The register below is used to enable catastrophic assertion into S5 state. This bit should always be set in all functional cases.

Table 20. Address Offset: 150Ch

Bit	Access	Default	Description
31	RWLO	0x0	**Policy Lock-Down Bit (CTENLOCK):** When written to 1, this bit prevents any more writes to this register
30:1	RO	0x0	**Reserved**

The thermal alert provides built in hysteresis, by having both a high and a low mark. An example of how it works is explained below:

- Both high and low marks are programmed to their correct values
  - Assume, for an example, the high value is 90°C, and the low value is 80°C.

- TS is enabled, and assume temperature is at ambient (50°C)
  - — Thus the alert signal is de-asserted.
- Temperature starts to rise as traffic flows through PCH
- Temperature reaches greater than 90°C
  - — Alert signal is asserted.
  - — Based on programming, a platform indication like SMI, or SCI can occur if SW had enabled such.
- Temperature reaches 95°C
  - — Alert signal remains asserted.
- Temperature starts to fall and reaches 85°C
  - — Alert signal remains asserted because it has not reached less than 80°C, which is the value to turn off alert.
- Temperature falls to less than 80°C
  - — Alert is turned off now since the temperature has fallen to the low value.
  - — Based on programming a platform indication like SMI, or SCI can occur if SW had enabled such.
- Temperature starts rising again and goes up to 85°C
  - — Alert remains off until temperature rises to the high mark of greater than 90°C.

An example of how SW can use the hysteresis would be to program a value for when the fans should be turned up, or cooling should be increased (90°C in example above), then allow the cooling to be sufficient that the extra cooling can be reduced (80°C). This prevents the PCH from oscillating around one temperature with the fans increasing/decreasing every few seconds. Using the hysteresis allows the fans to be on or off for much longer periods.

Table 21. Thermal Trip Points and Response (Typical)

Zone	Nominal Trip Points	Response
Catastrophic	TCatastrophic (fused catastrophic temp value)=119°C	Halt Operation required (for example, going to S5 State)
Hot	Threshold On = value set by OEM	SW Response recommended; (for example, turn fans up)
Cool	Threshold Off = value set by OEM	SW Response recommended; (for example, turn fans down)

4.3.5 Thermal Sensor Accuracy ($T_{accuracy}$)

The PCH thermal sensor accuracy is:

- ±5°C over the temperature range from 50°C to 110°C.
- ±7°C over the temperature range from 30°C to 50°C.
- ±10°C over the temperature range from -10°C to 30°C.
- The sensor itself is functional, from -40°C to 130°C, no accuracy is specified for temperature range beyond 110°C or below -10°C.

4.3.6 Thermal Trip Signal (PCHHOT_N)

The PCH provides PCHHOT_N signal to indicate that it has exceeded some temperature limit. The limit is set by BIOS. The temperature limit (programmed into the PHL register) is compared to the present temperature. If the present temperature is greater than the PHL value then the pin is asserted.

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

The PCH evaluates the temperature from the thermal sensor against the programmed temperature limit every one second.

5.0 Memory

This chapter contains information about the following:

- System Memory Interface
- Power Management

5.1 System Memory Interface

Memory controller can support DDR4 and LPDDR4x technologies. The memory system supports memory configuration 2x32 LPDDR4x, 4x32 LPDDR4x, 1x64 DDR4 and 2x64 DDR4.

The tables in this section describe the details of the supported configuration matrix.

Table 22. DDR Support Matrix

Feature	LPDDR4x	DDR4
DRAM Die Density	4 Gb 8 Gb 16 Gb	4 Gb 8 Gb 16 Gb
Maximum Data Rate (MT/s)	2933	2933
Channels	2 x 32 bits 4 x 32 bits	1 x 64 bits 2 x 64 bits
DPC[1]	Not Applicable	1
RPC[2]	2 (2933 MT/s)	2 (2933 MT/s)
Notes: 1. DPC = DIMMs per channel. 2. RPC = Rank Per Channel		

5.1.1 DRAM Channel Support Matrix and Signals Terminology

Table 23. LPDDR4x Sub-Channels Population Rules

Number of DRAMs	DRAM Type	Sub-Channel Population
2	x32	DRAM 0 is connected to Sub Channel A DRAM 1 is connected to Sub Channel B
4	x32	DRAM 0 is connected to Sub Channel A DRAM 1 is connected to Sub Channel B DRAM 2 is connected to Sub Channel C DRAM 3 is connected to Sub Channel D

Table 24. System Memory Interface Signals Terminology

Memory Type	DDR4 SoDIMM (Per Channel)	LPDDR4/4x Memory Down (All Channels)
Signal details		
Clock (CLK)	DDR_[1:0]_CLK[1:0]_DN, DDR_[1:0]_CLK[1:0]_DP	LP4x_[3:0]_CLK_DN, LP4x_[3:0]_CLK_DP
Control (CTRL)	DDR_[1:0]_CS[1:0]_N, DDR_[1:0]_ODT[1:0]	LP4x_0_CS0 LP4x_1_CS1 LP4x_2_CS0 LP4x_3_CS1
Clock Enable (CKE)	DDR_[1:0]_CKE[1:0]	LP4x_0_CKE0 LP4x_1_CKE1 LP4x_2_CKE0 LP4x_3_CKE1
Command (CMD)	DDR_[1:0]_MA[13:0], DDR_[1:0]_MA14_WE_N, DDR_[1:0]_MA15_CAS_N, DDR_[1:0]_MA16_RAS_N, DDR_[1:0]_BG[1:0], DDR_[1:0]_BA[1:0], DDR_[1:0]_ACT_N, DDR_[1:0]_PAR	LP4x_[3:0]_CA[5:0]
Alert	DDR_[1:0]_ALERT_N	N/A
Strobe	DDR_[1:0]_DQS[7:0]_DN, DDR_[1:0]_DQS[7:0]_DP	LP4x_[3:0]_DQS[3:0]_DN, LP4x_[3:0]_DQS[3:0]_DP
Data	DDR_[1:0]_DQ[63:0]	LP4x_[3:0]_DQ[31:0]
Reset	PMC_DRAM_RESET_N	PMC_DRAM_RESET_N
RCOMP	DDR_RCOMP[2:0]	LP4x_RCOMP[2:0]
Vref	DDR_[1:0]_VREF_CA	N/A
VTT	DDR_VTT_CTL	N/A

Table 25. SA Speed Enhanced Speed Steps (SA-GV) and Gear Mode Frequencies

	DDR max rate [MT/s]	SAGV-Low DDR CLK, Gear	SAGV-Mid DDR CLK, Gear	SAGV-High DDR CLK, Gear
DDR4	2133	2133,G2	2133,G2	2133,G2
	2400	2133,G2	2400,G2	2400,G2
	2666	2133,G2	2400,G2	2666,G2
	2933	2133,G2	2400,G2	2933,G2
LPDDR4/x	2133	2133,G2	2133,G2	2133,G2
	2400	2133,G2	2400,G2	2400,G2

continued...

	DDR max rate [MT/s]	SAGV-Low DDR CLK, Gear	SAGV-Mid DDR CLK, Gear	SAGV-High DDR CLK, Gear
	2666	2133,G2	2400,G2	2666,G2
	2933	2133,G2	2400,G2	2933,G2

Notes: 1. Intel® Pentium® Silver and Intel® Celeron® Processor supports dynamic gearing technology where the Memory Controller can run at 1:2 (Gear-2 mode)ratio of DRAM speed. 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. **Low**- Low frequency point, Min Power point. Characterized by low power, low BW, high latency. System will stay at this point during low to moderate BW consumption.

b. **Mid** - Max Bandwidths Point, this point is the max possible BW point, the DRAM freq limited by Silicon Configuration/BIOS/SPD. Characterized by moderate power and latency, high BW. This point intended for high GT and moderate-high IA BW

c. **High** - High Point, the minimum memory latency point, Characterized by high power, low latency, moderate BW. Only during IA performance workloads the system will to switch to this point and only in case this point can provide enough BW.

5.1.2 Memory Frequency

In all modes, the frequency of system memory is the lowest frequency of all memory modules placed in the system, as determined through the SPD registers on the memory modules. The system memory controller supports a single DIMM connector per channel. If DIMMs with different latency are populated across the channels, the BIOS will use the slower of the two latencies for both channels. For Dual-Channel modes both channels should have a DIMM connector populated. For Single-Channel mode, only a single channel can have a DIMM connector populated.

5.1.3 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 DDR4 command may be issued every 2 clocks.
  - — 1N indicates a new DDR4/LPDDR4x command may be issued every clock.

Table 26. DDR4 System Memory Timing Support

DRAM Device	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRP (ns)	CWL (tCK)	DPC	CMD Mode
DDR4	2933	22	13.75	13.75	9-12, 14,16,18,20	1	2N

Table 27. LPDDR4x System Memory Timing Support

DRAM Device	Mode	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRPpb (ns)	tRPab (ns)	WL (tCK) Set B
LPDDR4x	X32	2933	36	18	18	21	30

5.1.4 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

5.1.5 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.

5.1.6 Data Swapping

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

- byte (DQ+DQS) swapping between bytes in the same channel.
- bit swapping within specific byte.

5.1.7 DRAM Clock Generation

Every supported rank has a differential clock pair. There are a total of four clock pairs driven directly by the processor to DRAM.

5.1.8 DRAM Reference Voltage Generation

The memory controller has the capability of generating the LPDDR4X and DDR4 Reference Voltage (VREF) internally for both read and write operations. The Vref is trained during cold boot by advanced training procedures in order to provide the best channel margins.

5.2 Power Management

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

5.2.1 Disabling Unused System Memory Outputs

Any system memory (SM) interface signal that goes to a memory in which it is not connected to any actual memory devices (such as SoDIMM connector is unpopulated, or is single-sided) is tri-stated. The benefits of disabling unused SM signals are:

- Reduced leakage power consumption.
- Reduce unintended receiver operation due to unterminated transmission lines.
- When a given rank is not populated, the corresponding control signals (CLK_DP/ CLK_DN/CKE/ODT/CS) are not driven.

5.2.2 DRAM Power Management and Initialization

The processor implements extensive support for power management on the memory interface. Each channel drives 4 CKE pins, one per rank.

The CKE is one of the power-saving means. When CKE is off, 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 four 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 type of CKE power-down can be configured through PDWN_mode (bits 15:12) and the idle timer can be configured through PDWN_idle_counter (bits 11:0).

The different power-down modes supported are:

- **No power-down** (CKE disable)

- **Active power-down (APD):** This mode is entered if there are open pages when de-asserting CKE. In this mode the open pages are retained. Power-saving in this mode is the lowest. Power consumption of DDR is defined by IDD3P. Exiting this mode is fined by tXP – small number of cycles. For this mode, DRAM DLL should be on.
- **PPD/DLL-off:** In this mode the data-in DLLs on DDR are off. Power-saving in this mode is the best among all power modes. Power consumption is defined by IDD2P. Exiting this mode is defined by tXP, but also tXPDLL (10–20 according to DDR type) cycles until first data transfer is allowed. For this mode, DRAM DLL should be off.
- **Precharged power-down (PPD):** This mode is entered if all banks in DDR are pre-charged when de-asserting CKE. Power-saving in this mode is intermediate – better than APD, but less than DLL-off. Power consumption is defined by IDD2P. Exiting this mode is defined by tXP. The difference from APD mode is that when waking-up, all page-buffers are empty.) The LPDDR does not have a DLL. As a result, the power savings are as good as PPD/DDL-off but will have lower exit latency and higher performance.

The CKE 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 arrives to queues. The idle-counter begins counting at the last incoming transaction arrival.

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 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 – PPD/DLL-off with a low idle timer value
- 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 default value that BIOS configures in PM PDWN config register is 6080 – that is, PPD/DLL-off mode with idle timer of 0x80 (128 DCLKs). This is a balanced setting with deep power-down mode and moderate idle timer value.

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.

Initialization Role of CKE

During power-up, CKE is the only input to the SDRAM that has its level recognized (other than the reset pin) once power is applied. It should be driven LOW by the DDR controller to make sure the SDRAM components float DQ and DQS during power-up.

CKE signals remain LOW (while any reset is active) until the BIOS writes to a configuration register. Using this method, CKE is ensured to remain inactive for much longer than the specified 200 micro-seconds after power and clocks to SDRAM devices are stable.

Conditional Self-Refresh

During S0 idle state, system memory may be conditionally placed into self-refresh state when the processor is in package C3 or deeper power state. When entering the S3 – Suspend-to-RAM (STR) state or S0 conditional self-refresh, the processor IA core flushes pending cycles and then enters SDRAM ranks that are not used by the processor or graphics into self-refresh. The CKE signals remain LOW so the SDRAM devices perform self-refresh.

The target behavior is to enter self-refresh for package C3 or deeper power states as long as there are no memory requests to service.

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.

The processor IA core controller can be configured to put the devices in active powerdown (CKE de-assertion with open pages) or precharge power-down (CKE de-assertion with all pages closed).

Precharge power-down provides greater power savings but has a bigger performance impact, since all pages will first be closed before putting the devices in power-down mode.

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 refresh.

5.2.3 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 for the majority of the logic to reduce idle power while keeping all critical DDR pins such as CKE and VREF in the appropriate state.

In C7 or deeper power state, the processor internally gates VCCIO_EXT for all non-critical state to reduce idle power. In S3 or C-state transitions, the DDR does not go through training mode and will restore the previous training information.

5.2.4 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 operational margins using advanced mathematical models.

6.0 Graphics

This chapter contains information about Processor Graphics.

6.1 Processor Graphics

The processor graphics is based on Generation 11 (GEN11-LP GT1) graphics core architecture that enables substantial gains in performance and lower-power consumption over prior generations. Gen 11LP architecture supports up to 32 Execution Units (EUs) depending on the processor SKU.

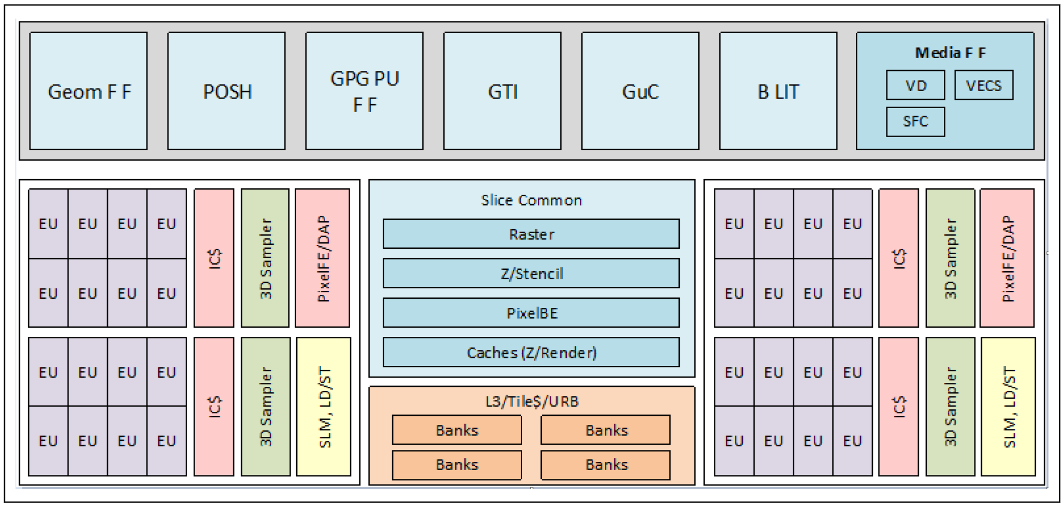
The processor graphics architecture delivers high dynamic range of scaling to address segments spanning low power to high power, increased performance per watt, support for next generation of APIs. Gen 11LP scalable architecture is partitioned by usage domains along Render/Geometry, Media, and Display. The architecture also delivers very low-power video playback. 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.

6.1.1 Graphic Features

Below are listed of features supported in the processor:

- Microsoft* DirectX 12 compliant.
- OpenCL™ 3.0, Vulkan* APIs.
- Dedicated FIVR for Graphics.
- Vtd, GT L3 size is 1280KB.
- Coarse Pixel Shading (CPS)
- Position Only Shading (POSh) is supported in Open GL™.
- POSh Tile Based Rendering (PTBR) is supported in Open GL™.
- GT Configuration **1x2x8, 1x4x6, 1x4x8**

Figure 5. Block Diagram



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

Gen 11LP implements multiple media video codecs in hardware.

6.1.3 Hardware Accelerated Video Decode and Encode

Gen 11LP implements a high-performance and low-power HW acceleration for video decoding and encoding operations.

Table 28. Hardware Accelerated Video Decode and Encode

Codec Format	Decode Level	Encode Level
H.265/HEVC	M10P @ L5.1 8b/10b Up-to 4kp60 (3480x2160); (420) Up-to 4kp30 (3480x2160); (444)	M10P @ L5.1 8b/10b Up-to 4kp60 (3480x2160); (420) Up-to 4kp30 (3480x2160); (444)
VP9	Profile 0,1,2,3 8b/10b Up-to 4kp60 (3480x2160); (420) Up-to 4kp30 (3480x2160); (444)	Profile 0,1,2,3 8b/10b Up-to 4kp60 (3480x2160); (420) Up-to 4kp30 (3480x2160); (444)
H.264	MP, HP, CBP L5.2 8b/ Up-to 4kp60 (3480x2160); (420)	MP, HP, CBP 8b/ Up-to 4kp60 (3480x2160); (420)
WMV9/VC1	SP ML/MP HL/AP L4 and up to 4Kp60 (3480x2160); AP L3 Up-to 1920x1080p24 AP L4 Up to 2048x1536p24 MP HL Up to 1920x1080p30 SP ML Up to 352x288p15	Not Supported

continued...

Codec Format	Decode Level	Encode Level
MPEG-2	1080p60 (MP@HL and MP@ML)	Not Supported
VC-1	AP L3 8b/ Up to 1080p30	Not Supported
JPEG/MJPEG	850Mpps (420), 640Mpps (422), 428Mpps (444)	800Mpps (420), 600Mpps (422)

Hardware Accelerated Transcoding

Transcoding is a combination of video decode 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:

- Low-power and low-latency AVC encoder for video conferencing and Wireless Display applications.
- Lossless memory compression for media engine to reduce media power.
- Low power Scaler and Format Converter.

7.0 Display

This chapter provides information on the following topics:

- General Capabilities
- Display Features
- Port Configuration
- SoC Display Architecture
- Display Interfaces
- High-Bandwidth Digital Content Protection (HDCP)
- Display Technologies

7.1 General Capabilities

- Three simultaneous displays (Pipes A,B,C)
- Seven planes and one cursor per pipe
- Audio streams per pipe to go to external ports
- HDR support for three planes per pipe
- VESA DSC compression support for A, B and C
- Post-DSC joining for resolutions that require more bandwidth than one pipe can support
- Pipe A optimized for low power
- Three combo PHY ports DSI/eDP/DP/HDMI
- AUX channels for Display ports and eDP
- Multi-stream support for Display ports
- PSR1, PSR2 and multi segmented operations, chip on glass for eDP

Table 29. Display Interface Versions

Interface	Versions
Display Serial Interface (DSI)	MIPI-DSI 1.3 @ 2.5 Gbps (up to seven inches trace)
Embedded Display Port (eDP*)	eDP 1.4b @ 5.4 Gbps
Display Port (DP)	DP 1.4 @ 8.1 Gbps (up to seven inches trace)
High Definition Multimedia Interface (HDMI)	HDMI 2.0b @ 5.94 Gbps (With Platform Level Shifters above HD resolution)
Maximum Resolution	4K2K

7.2 Features Supported

Feature	MIPI-DSI	eDP	DP	HDMI
Numbers of Ports	1 (1x4 and 1x8)	1 (x4)	3 (x4)[1]	3 (x4)[1]
Maximum Resolution	1x4: 4096 × 2160 @ 60 Hz (with DSC) 1x8: 4096 × 2160 @ 60 Hz (without DSC)	4096 x 2160@ 60 Hz	4096 x 2160@ 60 Hz	4096 x 2160 @ 60 Hz
Data Rate (Gbps per Lane)	2.5	5.4	8.1 (Type C with Re-Timer) 5.4 (Without Re-Timer)	5.94
Power gated during S0ix w/ display off	Yes	Yes	Yes	Yes
DRRS (Refresh reduction)	Yes (Panel command)	Yes	N/A	N/A
Self-Refresh with frame buffer in Panel	Yes (Command Mode)	Yes (PSR1,PSR2)	N/A	N/A
Content-Based back light control	DPST6.3/CABC LACE DPST	DPST6.3 LACE DPST	N/A	N/A
HDCP 2.2	N/A	N/A	Yes	Yes
PAVP	AES-encrypted buffer, plan control, panic attack			
HD Audio	N/A	N/A	Yes	Yes
Compressed Audio	N/A	N/A	Yes	Yes
DSC (Display Stream Compression)	Yes	Yes	Yes	No
Note: 1. SoC can support three HDMI or three DP ports.				

7.3 Port Configuration

Either Internal or External configuration is possible with each pipe. Only one configuration out of the list shown from below table is possible per port bases.

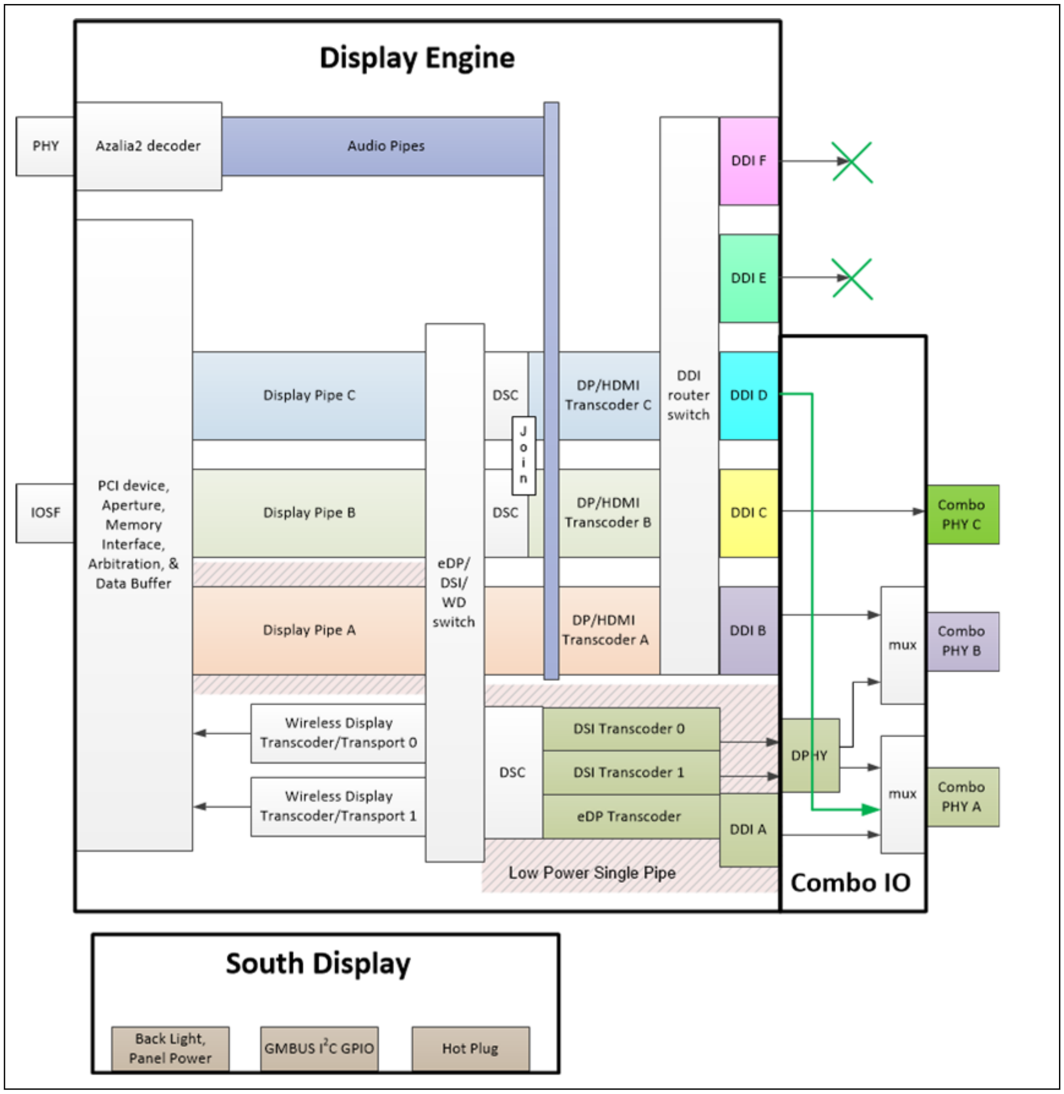
Table 30. Ports Availability

Combo PHY Port	Internal Port	External Display
DDI0 (Port A)	eDP MIPIA with DSI0	HDMI DP
DDI1 (Port B)	MIPIB with DSI1	HDMI DP
DDI2 (Port C)	NA	HDMI DP

Notes:
- PSR2 supported only on DDI0 in single eDP mode
- Intel® Pentium® Silver and Intel® Celeron® Processor supports single internal display only

7.4 SoC Display Architecture

Figure 6. Display Engine



7.5 Display Interfaces

Table 31. Display DDI Data and Clock Signals

Package Pin	Dir.	eDP	MIPI DSI	DP	HDMI
DISP_RCOMP	N/A	Common RCOMP for all PHYs			
DDI0_AUXP DDI0_AUXN	I/O	eDP Auxiliary Channel (AUX_CH)	MIPI0 Data 0	DP0 Auxiliary Channel (AUX_CH)	NC
DDI0_TXN0 DDI0_TXP0	I/O	eDP Main Link, Lane 0 (ML_Lane 0)	MIPI0 Data 1	DP0 Main Link, Lane 0 (ML_Lane 0)	TMDS0 Data2

continued...

Package Pin	Dir.	eDP	MIPI DSI	DP	HDMI
DDI0_TXN1 DDI0_TXP1	O	eDP Main Link, Lane 1 (ML_Lane 1)	MIPI0 Data 2	DP0 Main Link, Lane 1 (ML_Lane 1)	TMDS0 Data1
DDI0_TXN2 DDI0_TXP2	O	eDP Main Link, Lane 2 (ML_Lane 2)	MIPI0 Clock	DP0 Main Link, Lane 2 (ML_Lane 2)	TMDS0 Data 0
DDI0_TXN3 DDI0_TXP3	O	eDP Main Link, Lane 3 (ML_Lane 3)	MIPI0 Data 3	DP0 Main Link, Lane 3 (ML_Lane 3)	TMDS0 Clock
DDI1_AUXN DDI1_AUXP	I/O	NC	MIPI1 Data 0	DP1 Auxiliary Channel (AUX_CH)	NC
DDI1_TXN0 DDI1_TXP0	I/O	NC	MIPI1 Data 1	DP1 Main Link, Lane 0 (ML_Lane 0)	TMDS1 Data2
DDI1_TXN1 DDI1_TXP1	O	NC	MIPI1 Data 2	DP1 Main Link, Lane 1 (ML_Lane 1)	TMDS1 Data1
DDI1_TXN2 DDI1_TXP2	O	NC	MIPI1 Clock	DP1 Main Link, Lane 2 (ML_Lane 2)	TMDS1 Data0
DDI1_TXN3 DDI1_TXP3	O	NC	MIPI1 Data 3	DP1 Main Link, Lane 3 (ML_Lane 3)	TMDS1 Clock
DDI2_AUXN DDI2_AUXP	I/O	NC	NC	DP2 Auxiliary Channel (AUX_CH)	NC
DDI2_TXN0 DDI2_TXP0	I/O	NC	NC	DP2 Main Link, Lane 0 (ML_Lane 0)	TMDS2 Data2
DDI2_TXN1 DDI2_TXP1	O	NC	NC	DP2 Main Link, Lane 1 (ML_Lane 1)	TMDS2 Data1
DDI2_TXN2 DDI2_TXP2	O	NC	NC	DP2 Main Link, Lane 2 (ML_Lane 2)	TMDS2 Data0
DDI2_TXN3 DDI2_TXP3	O	NC	NC	DP2 Main Link, Lane 3 (ML_Lane 3)	TMDS2 Clock

Table 32. Pin Mapping for Display Control Signals

Display Signals	Dir.	Description	Usage Model			
			eDP/DP/HDMI Port 0, DP/ HDMI Port 1, DP/HDMI Port 2	DSI Dual Link Port 0+1, DP/HDMI Port 2	DSI Port 0, DP/HDMI port 1, DP/HDMI Port 2 (Note1)	DSI Port 0, DP/HDMI Port 2 (Note2)
DDI0_HPD	I/O	DDI0 Hot Plug Detection	DDI0 eDP HPD			
DDI1_HPD	I/O	DDI1 Hot Plug Detection	DP/HDMI HPD		DDI1 DP/HDMI HPD	
eDP_VDDEN	I/O	Panel main power enable	VDD enable for eDP	MIPI DSI power enable AVDD	MIPI DSIPower Enable AVDD	MIPI DSI Power Enable AVDD
DDI2_HPD	I	Dedicated DDI2 Hot Plug Detection	DDI2 DP/HDMI HPD	DDI2 DP/HDMI HPD	DDI2 DP/HDMI HPD	DDI2 DP/HDMI HPD
eDP_BKLTEN	O	Panel backlight enable	eDP Backlight Enable	MIPI DSI Backlight Enable	MIPI DSI Backlight Enable	MIPI DSI Backlight Enable

continued...

Display Signals	Dir.	Description	Usage Model			
			eDP/DP/HDMI Port 0, DP/ HDMI Port 1, DP/HDMI Port 2	DSI Dual Link Port 0+1, DP/HDMI Port 2	DSI Port 0, DP/HDMI port 1, DP/HDMI Port 2(Note1)	DSI Port 0, DP/HDMI Port 2 (Note2)
eDP_BKLTCTL	O	Panel backlight control	eDP Backlight Control	MIPI DSI Backlight Control	MIPI DSI Backlight Control	MIPI DSI Backlight Control
DDI0_DDC_SCL	I/O	Panel reset or DDC clock for HDMI 0 or DSI panel secondary power (AVEE)	Control Clock 0	DSI power AVEE	MIPI0 Reset	DSI power AVEE
DDI0_DDC_SDA	I/O	DDC data for HDMI 0 or DSI panel secondary power (AVDD)	Control Data 0	DSI power AVDD	DSI power AVEE	DSI power AVEE
DDI1_DDC_SCL	I/O	Panel reset or DDC clock for HDMI 1	Control Clock 1	Dual Link Mode Reset	Control Clock 1	MIPI0 Reset
DDI1_DDC_SDA	I/O	DDC data for HDMI 1	Control Data 1		Control Data 1	
DDI2_DDC_SCL	I/O	Dedicated DDI2 DDC Clock for HDMI or DP++	Control Clock 2	Control Clock 2	Control Clock 2	Control Clock 2
DDI2_DDC_SDA	I/O	Dedicated DDI2 DDC Data for HDMI and DP++	Control Data 2	Control Data 2	Control Data 2	DDI2 DDC Data
MDSI_DE_TE_1	I/O	Tearing Effect from MIPI Panel 0		Dual link MIPI TE	MIPI0 TE	MIPI0 TE
MDSI_DE_TE_2	I/O	Tearing Effect from MIPI Panel 1		Dual link MIPI TE		

NOTES

1. Usage that supports DP/HDMI port 1. Cannot support dual independent DSI analog/secondary power controls.
2. Usage that supports dual independent DSI analog/secondary power controls. Cannot support DP/HDMI port 1
3. MIPI Reset is not validated on Intel RVP
4. PNL_VDDEN, PNL_BLKLTEN, PNL_BKLTCTL can be left no connect if neither eDP or MIPI-DSI is not used.

7.6 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 2.2 and 1.4

for 4k Premium content protection over wired displays (HDMI and DisplayPort). The HDCP 1.4/2.2 keys are integrated into the processor and customers are not required to physically configure or handle the keys.

7.7 Display Technologies

This section contains information about the following:

- DisplayPort
- High-Definition Multimedia Interface (HDMI)
- Embedded DisplayPort (eDP*)
- MIPI DSI
- More Features of Display Controller
- Integrated Audio

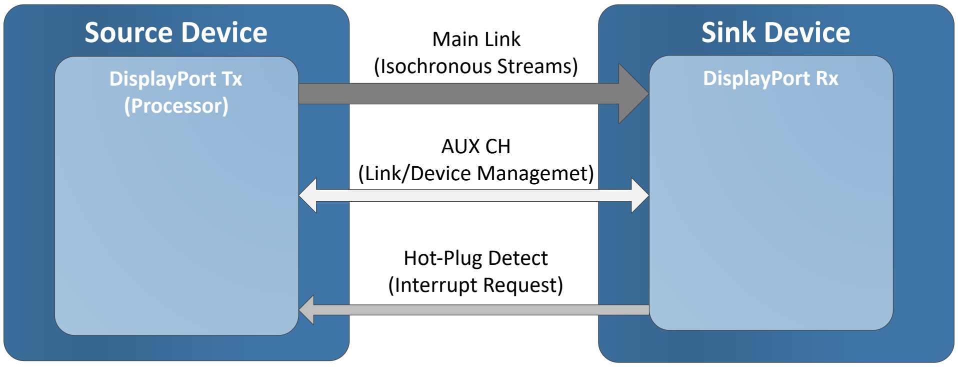
7.7.1 DisplayPort

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 (4 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 bidirectional channel used for link management and device control. The Hot-Plug Detect (HPD) signal serves as an interrupt request for the sink device.

The processor is designed in accordance to VESA DisplayPort specification.

Figure 7. DisplayPort Overview



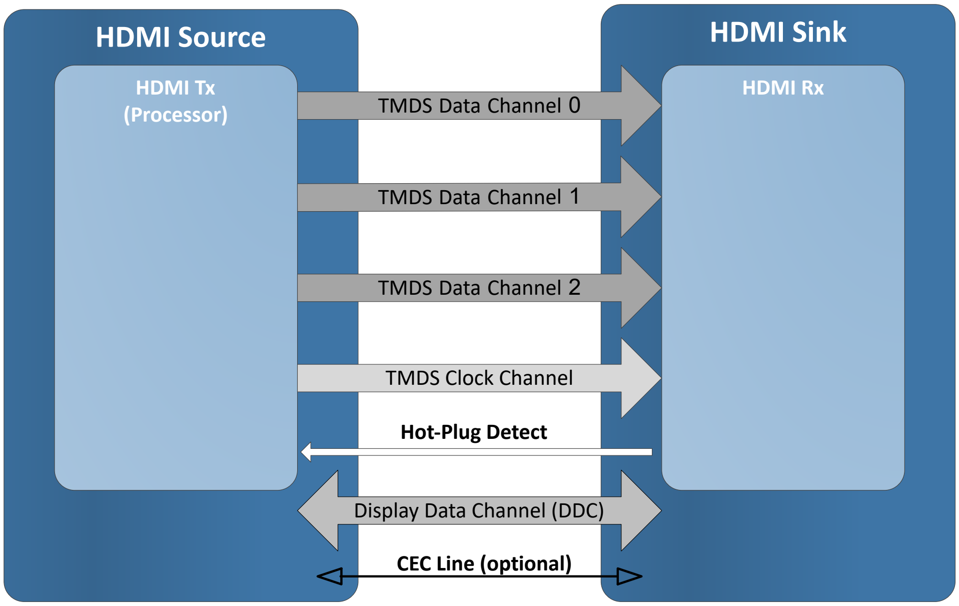
7.7.2 High-Definition Multimedia Interface (HDMI)

The High-Definition Multimedia Interface (HDMI) is provided for transmitting uncompressed 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 also needs an external component.

HDMI includes three separate communications channels: TMDS, Digital Display Channel (DDC), 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. These channels are used to carry video, audio, and auxiliary data. In addition, HDMI carries a VESA DDC. The DDC 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. The video pixel clock is transmitted on the TMDS clock channel and is used by the receiver for data recovery on the three data channels. The digital display data signals driven natively through the PCH are AC coupled and needs level shifting to convert the AC coupled signals to the HDMI compliant digital signals. The processor HDMI interface is designed in accordance with the High-Definition Multimedia Interface.

Figure 8. HDMI Overview



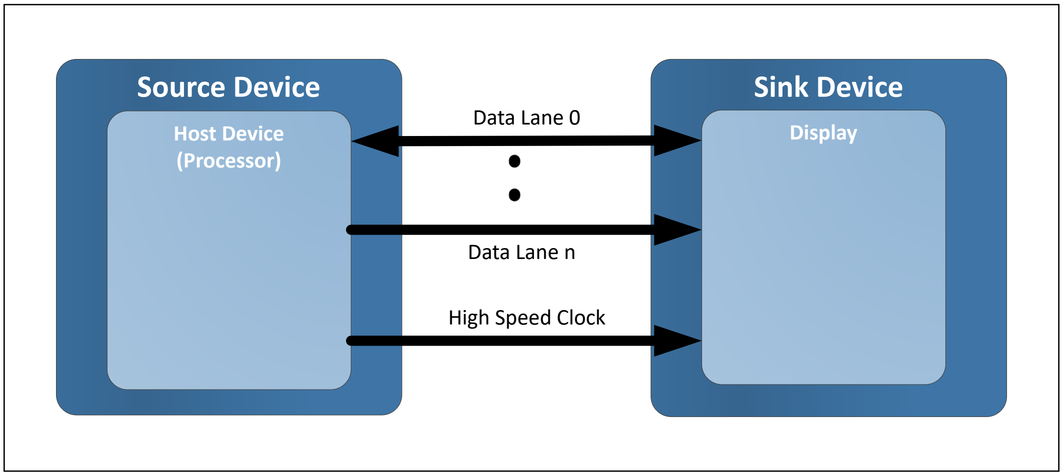
7.7.3 Embedded DisplayPort (eDP*)

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 a Main Link, Auxiliary channel, and an optional Hot-Plug Detect signal.

7.7.4 MIPI DSI

Display Serial Interface (DSI) specifies the interface between a host processor and peripheral such as display module. DSI is a high speed and high performance serial interface that offers efficient and low power connectivity between the processor and display module. The processor supports single or dual link interface.

Figure 9. MIPI DSI Overview



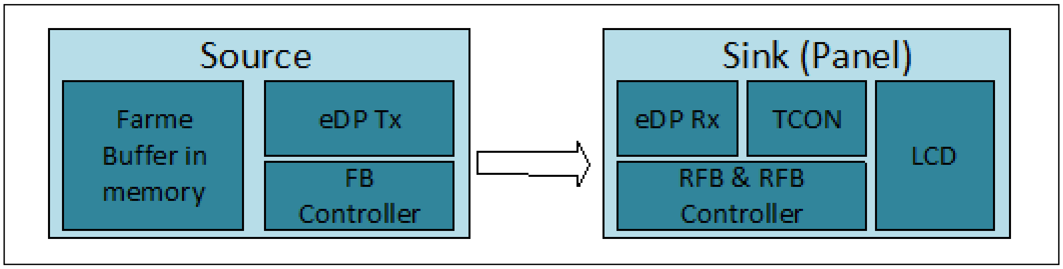
7.7.5 More Features of Display Controller

Panel Self Refresh (PSR)

PSR is an eDP feature that allows refresh to stop when the image is unchanging. Panel support is required for PSR1 and PSR2 capabilities. Display Engine (DE) can disable the eDP link and stop reading pixels from memory. The panel stores the unchanging image in it’s Remote Frame Buffer (RFB).

DE tracks image changes and automatically enters and exits PSR. Panel Self Refresh 2 (PSR2) adds several enhancements, including selective update.

Figure 10. Panel Self Refresh Diagram



7.7.6 Integrated Audio

HDMI and DisplayPort interfaces carry audio along with video.

The processor supports Three High Definition audio streams on Three digital ports simultaneously (the DMA controllers are in PCH). The integrated audio processing is performed by the PCH, and delivered to the CPU using the Serial Data Output (SDO) and Bit Clock (BCLK) signals. The Serial Data Input (SDI) is used to carry responses back to the PCH

This HDA interface is not available for use with external CODECs.

Table 33. 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, 192 kHz/24 bit, 6 Channel	Yes	Yes
Dolby* TrueHD, DTS-HD Master Audio (Lossless Blu-Ray Disc Audio Format)	Yes	Yes

8.0 Imaging

The Imaging Processing Unit (IPU) in SoC is the IPU6SE. IPU uses MIPI CSI to get data from the cameras. IPU supports up to four total cameras (three concurrent) with eight data lanes and four clock lanes of MIPI CSI over DPHY1.2.

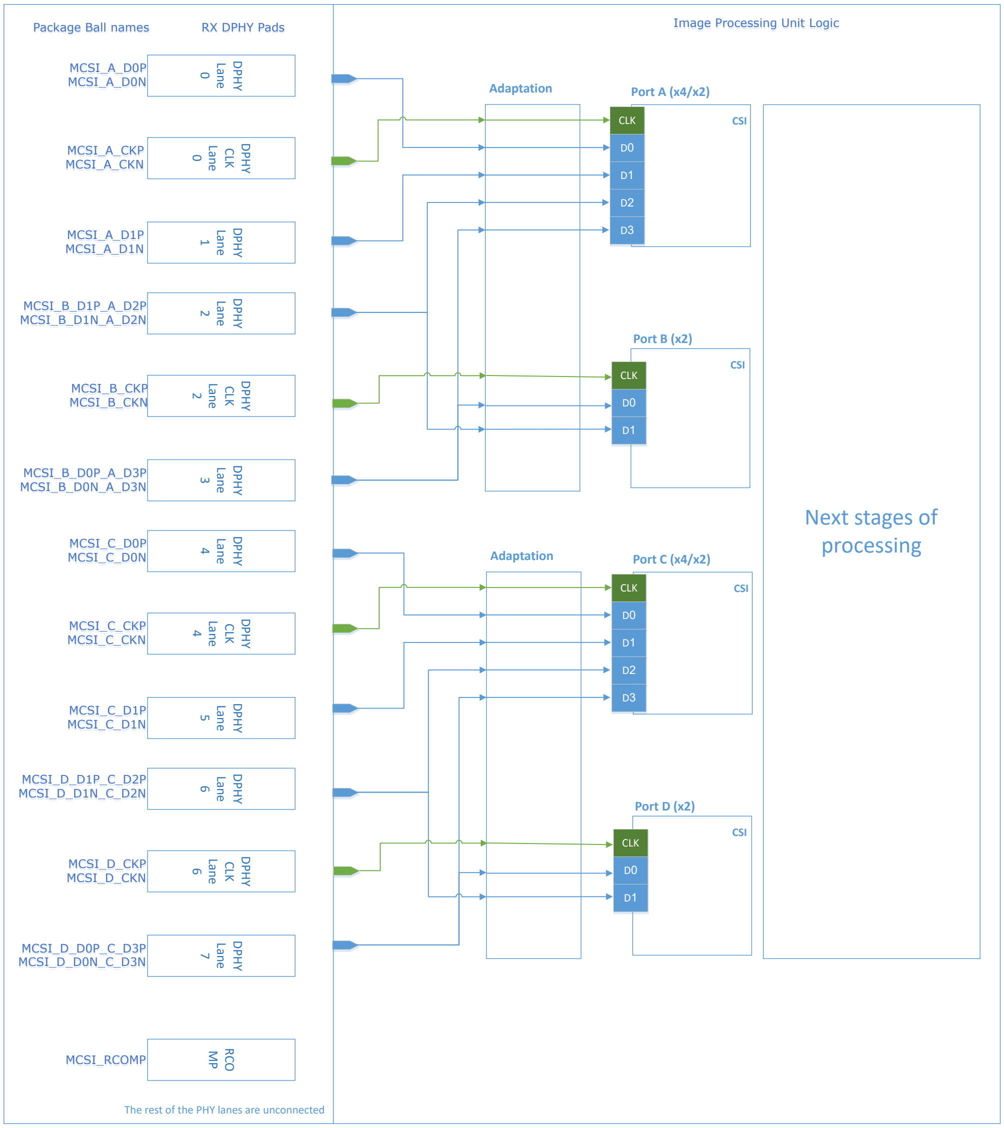
Table 34. SoC Imaging Feature Support

Feature	Support
Video recording • Single camera • Multiple cameras	Yes 1080p@30 2x 720p@30
Slow-motion video recording	Yes 720p@60
Video conferencing	Yes, Skype*/Lync*
Still capture from preview	Yes, 12mp
Still capture during video	Yes, up to 1080@30 +12mp SDV
Enhanced video capabilities – HDR	Yes
Analytics / **C**omputational **I**maging • Still capture CI • Video recording CI	Yes, Intel/OEM CI • Multi-frame CI ULL/HDR • Panorama

8.1 DPHY and CSI Controller Support

IPU6SE supports DPHY 1.2 with a total of 8 data lanes and 4 clock lanes. Up to 4 data lanes are supported per camera.

The following diagram shows the connectivity between the DPHY and the CSI controllers. The adaptation layer allows the 8 data lanes to be used as two X4 cameras or four x2 cameras.



8.1.1 Camera Connectivity Combinations

Using 8 DPHY1.2 data lanes, SoC can connect to up to four camera modules, and use up to three active camera modules out of the four, concurrently.

8.1.2 Signal Description

MIPI CSI Port A and Port C can be configured as x4.

Table 35. Camera Signals

Camera Signals	Description
MCSI_A_CKP	Differential clock (Port A)
MCSI_A_CKN	Differential clock (Port A)
MCSI_A_D0P	Lane 0 Differential data (Port A)
MCSI_A_D0N	Lane 0 Differential data (Port A)
MCSI_A_D1P	Lane 1 Differential data (Port A)
MCSI_A_D1N	Lane 1 Differential data (Port A)
MCSI_B_D1P_A_D2P	Differential data (Lane 1 Port B/Lane 2 Port A)
MCSI_B_D1N_A_D2N	Differential data (Lane 1 Port B/Lane 2 Port A)
MCSI_B_D0P_A_D3P	Differential data (Lane 0 Port B/Lane 3 Port A)
MCSI_B_D0N_A_D3N	Differential data (Lane 0 Port B/Lane 3 Port A)
MCSI_B_CKP	Differential clock (Port B)
MCSI_B_CKN	Differential clock (Port B)
MCSI_C_CKP	Differential clock (Port C)
MCSI_C_CKN	Differential clock (Port C)
MCSI_C_D0P	Lane 0 Differential data (Port C)
MCSI_C_D0N	Lane 0 Differential data (Port C)
MCSI_C_D1P	Lane 1 Differential data (Port C)
MCSI_C_D1N	Lane 1 Differential data (Port C)
MCSI_D_D1P_C_D2P	Differential data (Lane 1 Port D/Lane 2 Port C)
MCSI_D_D1N_C_D2N	Differential data (Lane 1 Port D/Lane 2 Port C)
MCSI_D_D0P_C_D3P	Differential data (Lane 0 Port D/Lane 3 Port C)
MCSI_D_D0N_C_D3N	Differential data (Lane 0 Port D/Lane 3 Port C)
MCSI_D_CKP	Differential clock (Port D)
MCSI_D_CKN	Differential clock (Port D)
MCSI_RCOMP	Compensation Resistor

8.2 Image Processing Capabilities

IPU6SE fixed function pipe supports the following functions:

- Black level correction.
- White balance.
- Color matching.
- Lens shading (vignette) correction.
- Color crosstalk (color shading)correction.
- Dynamic defect pixel replacement.
- Autofocus-pixel (PDAF) replacement.

- High quality demosaic.
- Scaling and format conversion.
- Temporal noise reduction running on Intel graphics.

9.0 Pin Strap

The following signals are used for static configuration. They are sampled at the rising edge of PMC_DSW_PWROK, PMC_RSMRST_N, or PMC_PCH_PWROK to select configuration and then revert later to their normal usage. To invoke the associated mode, the signal should meet both set up and hold time of 1us, with respect to the rising edge of the sampling signal.

Table 36. Pin Straps

Signal	Usage	When Sampled	Comment
GP_C01	Top Swap Override	Rising edge of PMC_PCH_PWROK	The strap has a 20 kohm ± 30% internal pull-down. 0 = **Disable** "Top Swap" mode. (Default) 1 = **Enable** "Top Swap" mode. This inverts an address on access to SPI and firmware hub, so the processor believes it fetches the alternate boot block instead of the original boot-block. PCH will invert A16 (default) for cycles going to the upper two 64-KB blocks in the FWH or the appropriate address lines (A16, A17, or A18) as selected in Top Swap Block size soft strap. *Notes:* 1. The internal pull-down is disabled after PCH_PWROK is high. 2. Software will not be able to clear the Top Swap bit until the system is rebooted. 3. The status of this strap is readable using the Top Swap bit (Bus0, Device31, Function0, offset DCh, bit4). This signal is in the primary well.
GP_C02	No Reboot	Rising edge of PMC_PCH_PWROK	The strap has a 20 kohm ± 30% internal pull-down. 0 = **Disable** "No Reboot" mode. (Default) 1 = **Enable** "No Reboot" mode (PCH will disable the TCO Timer system reboot feature). This function is useful when running ITP/XDP. *Notes:* 1. The internal pull-down is disabled after PCH_PWROK is high. 2. This signal is in the primary well.
GP_C08/UART0_RXD	TLS Confidentiality	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-down. 0 = **Disable** Intel® CSE Crypto Transport Layer Security (TLS) cipher suite (no confidentiality). (Default) 1 = **Enable** Intel® CSE Crypto Transport Layer Security (TLS) cipher suite (with confidentiality). *Notes:* 1. The internal pull-down is disabled after PMC_RSMRST_N de-asserts. 2. This signal is in the primary well.
GP_C09/UART0_TXD	eSPI Disable	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-down. 0 = **Enable** eSPI. (Default) 1 = **Disable** eSPI. *Notes:* 1. The internal pull-down is disabled after PMC_RSMRST_N de-asserts. 2. This signal is in the primary well.

continued...

Signal	Usage	When Sampled	Comment
GP_C10/UART0_RTS_N	Reserved	Rising edge of PMC_RSMRST_N	External pull-up is required. Recommend 100 K if pulled up to 3.3 V or 75 K if pulled up to 1.8 V. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling. There is no internal termination.
GP_C13/UART1_TXD	CPUNSSC Clock Frequency	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-down. 0 = 38.4 MHz clock (direct from crystal) (default) 1 = 19.2 MHz clock (derived from 38.4 MHz crystal) *Notes:* 1. The internal pull-down is disabled after PMC_RSMRST_N de-asserts. 2. When used as PCHHOT# and strap low, a 150 K pull-up is needed to ensure it does not override the internal pull-down strap sampling. 3. This signal is in the primary well.
GP_D08/ SIO_SPI2_CS0_N/ UART0A_RXD	Reserved	Rising edge of PMC_RSMRST_N	External pull-up is required. Recommend 100 K if pulled up to 3.3 V or 75 K if pulled up to 1.8 V. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling. There is no internal termination.
GP_D09/SIO_SPI2_CLK/ SIO_UART0A_TXD	Reserved	Rising edge of PMC_RSMRST_N	External pull-up is required. Recommend 100 K if pulled up to 3.3 V or 75 K if pulled up to 1.8 V. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling. There is no internal termination.
GP_A15	Flash Descriptor Security Override	Rising edge of PMC_PCH_PWROK	This strap has a 20 kohm ± 30% internal pull-down. 0 = **Enable** security measures defined in the Flash Descriptor. (Default) 1 = **Disable** Flash Descriptor Security (override). This strap should only be asserted high using external Pull-up in manufacturing/debug environments ONLY. *Notes:* 1. The internal pull-down is disabled after PCH_PWROK is high. 2. This signal is in the primary well.
GP_E06/IMGCLKOUT_3	Reserved	Rising edge of PMC_RSMRST_N	External pull-up is required. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling. There is no internal termination.
GP_E14/ DDI0_DDC_SDA	Used for BSSB_LS0(1.8V) or the display GMBus(3.3V)	Rising edge of PMC_RSMRST_N	This strap has 20 K internal pull-down. 0 = GP_E13/GP_E14 pins at 1.8 V 1 = GP_E13/GP_E14 pins at 3.3 V *Note:* This signal is in the primary well.
GP_E16/ DDI1_DDC_SDA	Used for BSSB_LS1(1.8V) or the display GMBus(3.3V)	Rising edge of PMC_RSMRST_N	This strap has 20 K internal pull-down. 0 = GP_E15/GP_E16 pins at 1.8 V 1 = GP_E15/GP_E16 pins at 3.3 V *Note:* This signal is in the primary well.

continued...

Signal	Usage	When Sampled	Comment
GP_E18/ DDI2_DDC_SDA	Used for BSSB_LS2(1.8V) or the display GMBus(3.3V)	Rising edge of PMC_RSMRST_N	This strap has 20 K internal pull-down. 0 =GP_E17/GP_E18 pins at 1.8 V 1 = GP_E17/GP_E18 pins at 3.3 V *Note:* This signal is in the primary well.
GP_E12/IMGCLKOUT_4	Used for BSSB_LS3(1.8V) or the display GMBus(3.3V)	Rising edge of PMC_RSMRST_N	This strap has 20 K internal pull-down. 0 = GP_E11/GP_E12 pins at 1.8 V 1 = GP_E11/GP_E12 pins at 3.3 V *Note:* This signal is in the primary well.
GP_D10	RSVD	Rising edge of PMC_RSMRST_N	This strap has 20 K internal pull-down. Do not pull this pin high on board.
DBG_PMODE	Reserved	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-up. This strap should sample high. There should NOT be any on-board device driving it to opposite direction during strap sampling. *Notes:* 1. The internal pull-up is disabled after PMC_RSMRST_N de-asserts. 2. This signal is in the primary well.
GP_DSW07	Reserved	Rising edge of PMC_DSW_PWROK	This strap has a 20 kohm ± 30% internal pull-down. This strap should sample LOW. There should NOT be any on-board device driving it to opposite direction during strap sampling. *Notes:* 1. The internal pull-down is disabled after DSW_PWROK is high. 2. This signal is in the DSW well.
GP_E00/IMGCLKOUT_0	XTAL Frequency Selection	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-down. This strap should not be pulled high since 24 MHz crystal is not supported on the PCH. 0 = 38.4 MHz/19.2 MHz(default) 1 = 24 MHz *Notes:* 1. The internal pull-down is disabled after PMC_RSMRST_N de-asserts. 2. This signal is in the primary well.
GP_E22/CNV_RGI_DT	M.2 CNVi Mode Select	Rising edge of PMC_RSMRST_N	A weak external pull-up is required. 0 = Integrated CNVi enabled. 1 = Integrated CNVi disabled. *Note:* When a RF companion chip is connected to the PCH CNVi interface. There is no internal termination.
GP_D11/SPI2_MOSI/ UART0A_CTS_N	eSPI Flash Sharing Mode	Rising edge of PMC_RSMRST_N	This strap has a 20 kohm ± 30% internal pull-down. 0 = Master Attached Flash Sharing (MAFS) is enabled. (Default) 1 = Slave Attached Flash Sharing (SAFS) is enabled. *Note:* 1. The internal pull-down is disabled after PMC_RSMRST_N de-asserts. This signal is in the primary well.
INTRUDER_N[2]	SPI Voltage Configuration	SRTCRST_N	There is no internal pull-up or pull-down on the signal. An external pull-up / pull-down is required. 0 = SPI operation voltage is 3.3 V (10 kohm pull-down to GND)

continued...

Signal	Usage	When Sampled	Comment
			1 = SPI operation voltage is 1.8 V (1 Mohm pull-up to VCCRTC)
CFG_00	EAR	-	1 = (Default) Normal Operation;. 0 = Reserved.
CFG_04	eDP Presense	-	Embedded Display Port Presence Strap 1= (default) disabled. 0=enabled.

NOTES

1. CFG signals have a default value as '1'.
2. INTRUDER_N is exclusively used for SPI Voltage Configuration. Chassis Intrusion Detection is not supported.

10.0 General Purpose Input and Output (GPIO)

The PCH General Purpose Input/Output (GPIO) signals are grouped into multiple groups (such as GP_A, GP_B, and so on) and are powered by either the PCH Primary well or Deep Sleep well.

The high level features of GPIO:

- Configurable 3.3 V or 1.8 V voltage (except for GP_F and GP_DSW groups)
- Configurable as an input or output signal.
- SCI (GPE) and interrupt capable on all GPIOs
- NMI and SMI capability capable (on selected GPIOs).
- PWM, Serial Blink capable (on selected GPIOs).
- Programmable hardware debouncer (on GP_DSW03/PMC_PWRBTN_N pin)

Table 37. Acronyms

Acronyms	Description
GPI	General Purpose Input
GPO	General Purpose Output
GP	General Purpose I/O in Primary Well
GP_DSW	General Purpose I/O in Deep Sleep Well
in	Input to SOC
out	Output from SOC
inout	Input and Output
od	Output open drain
iod	Input and output open drain

10.1 Signal Description

For GPIO signals and its description, download the pdf, click 📎 on the navigation pane and refer the spreadsheet, **633935-GPIO-Signals.xlsx**.

10.2 Functional Description

This section provides information on the following topics:

- Configurable GPIO Voltage
- GPIO Buffer Impedance Compensation via SD3_RCOMP
- Programmable Hardware Debouncer
- Integrated Pull-ups and Pull-downs

- SCI / SMI# and NMI
- Timed GPIO (TIME_SYNC)
- GPIO Blink (BK) and Serial Blink (SBK)
- Interrupt / IRQ via GPIO Requirement
- GPIO Ownership
- Native Function and TERM Bit Setting
- Virtual GPIO (vGPIO)

10.2.1 Configurable GPIO Voltage

Except for all pads in GPIO F, GPIO S, and GP_DSW groups, all other GPIO pads support per-pad configurable voltage, which allows control selection of 1.8 V or 3.3 V for each pad. The configuration is done via soft straps.

Before soft straps are loaded, the default voltage of each pin depends on its default as input or output.

- **Input**: 1.8 V level with 3.3 V tolerant.
- **Output**: the pin drives 3.3 V via a ~20 K pull-up. With this, any 1.8 V device must be capable of taking 20K pull-up to 3.3 V.

WARNING

GPIO pad voltage configuration must be set correctly depending on device connected to it; otherwise, damage to the PCH or the device may occur.

NOTES

- GPIO F and S groups support 1.8 V only.
- GP_DSW group supports 3.3 V only.

10.2.2 GPIO Buffer Impedance Compensation via SD3_RCOMP

All GPIO buffers require impedance compensation for 1.8 V and 3.3 V operation. The impedance compensation is done via the SD3_RCOMP signal. Therefore, SD3_RCOMP signal must have a precision pull down resistor of 200 Ohm (1 %) to GND (regardless of SDXC being used or not). Without proper impedance compensation, the GPIO buffers, including the muxed native functions, may not operate as expected.

10.2.3 Programmable Hardware Debouncer

Hardware debounce capability is supported on GP_DSW03/PMC_PWRBTN_N pad. The capability can be used to filter signal from switches and buttons if needed.

The period can be programmed from 8 to 32768 times of the RTC clock by programming the Pad Configuration DW2 register. At 32 kHz RTC clock, the debounce period is 244 us to 1 s.

10.2.4 Integrated Pull-ups and Pull-downs

All GPIOs have programmable internal pull-up/pull-down resistors which are off by default. The internal pull-up/pull-down for each GPIO can be enabled by BIOS programming the corresponding PAD_CFG_DW0 register.

10.2.5 SCI / SMI# and NMI

SCI capability is available on all GPIOs, while SMI and NMI capability is available on only select GPIOs.

Below are the PCH GPIOs that can be routed to generate SMI# or NMI:

- GP_B14, GP_B20, GP_B23
- GP_C[23:22]
- GP_D[6:2]
- GP_E[8:0], GP_E[16:13]

10.2.6 Timed GPIO (TIME_SYNC)

The PCH supports 2 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 PCH Always Running Timer (ART) time in the Time Capture register. The GPIO input event must be asserted for at least 2 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 2 crystal oscillator clocks period.

Timed GPIO supports event counter. When Timed GPIO is configured as input, event counter increments by 1 for every input event triggered. When Timed GPIO is configured as output, event counter increments by 1 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.

10.2.7 GPIO Blink (BK) and Serial Blink (SBK)

Certain GPIOs are capable of supporting blink and serial blink, indicated as BK and SBK respectively in the GPIO Signals table above. The BK and SBK are implemented as native functions muxed on the selected GPIOs. To enable BK or SBK on a GPIO having the capability, BIOS needs to select the assigned native function for BK or SBK on the GPIO.

10.2.8 Interrupt / IRQ via GPIO Requirement

A GPIO, as an input, can be used to generate an interrupt/IRQ to the PCH. In this case, it is required that the pulse width on the GPIO must be at least 4 us for the PCH to recognize the interrupt.

10.2.9 Native Function and TERM Bit Setting

Certain native function signals that are multiplexed onto GPIO pins support dynamic termination override, which allows the native controller to dynamically control the integrated pull-up / pull-down resistors on the signals. For those native function signals, when used, software must program the TERM bit field in the corresponding GPIO’s Pad Configuration DW1 to 1111b.

The table below shows the native function signals that support dynamic termination override.

Table 38. Native Function Signals Supporting Dynamic Termination Override

Native Function	Signal with Dynamic Termination Override
Intel® HD Audio	HDA_SDI[0:1] HDA_SDO HDA_SYNC I2S[2:0]_SCLK I2S[2:0]_SFRM I2S[2:0]_RXD DMIC_DATA_[1:0] SNDW1_DATA
SDXC (SD Card)	SD_SDIO_CMD SD_SDIO_D[3:0] SD_SDIO_CLK

10.2.10 Virtual GPIO (vGPIO)

vGPIO is a special type of GPIO implemented in the PCH for a specific functionality. vGPIO is not a physical GPIO; the signal is not balled out on the package. Programming the vGPIO is similar to programming a physical GPIO.

The PCH implements vGPIO39 (in GPIO community 1), which is specifically used for SD card detection as an interrupt generation. If the PCH integrated SD card is utilized, in conjunction of the SD_CD# pin to be used as card detect, a physical GPIO pin is required for interrupt generation. vGPIO39 is intended to replace the need for this addition physical GPIO if desired. SW needs to program the vGPIO accordingly to enable this functionality.

11.0 PCH Electrical Specification

This chapter contains the DC and AC characteristics for the PCH.

11.1 Absolute Maximum Ratings

Table 39. PCH Absolute Power Rail Maximum and Minimum Ratings

Voltage Rail	Minimum Limits	Maximum Limits
1.05 V	-0.5 V	1.4 V
1.8 V	-0.5 V	2.3 V
3.3 V	-0.7 V	3.7 V

PCH Absolute Power Rail Maximum and Minimum Ratings specifies absolute maximum and minimum ratings. At conditions outside functional operation condition limits, but within absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. If a device is returned to conditions within functional operation limits after having been subjected to conditions outside these limits (but within the absolute maximum and minimum ratings) the device may be functional, but with its lifetime degraded depending on exposure to conditions exceeding the functional operation condition limits.

At conditions exceeding absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. Moreover, if a device is subjected to these conditions for any length of time, it will likely either not function or its reliability will be severely degraded when returned to conditions within the functional operating condition limits.

Although the PCH contains protective circuitry to resist damage from Electrostatic Discharge (ESD), precautions should always be taken to avoid high static voltages or electric fields.

11.2 General DC Characteristics

Table 40. Single-Ended Signal DC Characteristics as Inputs or Outputs

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Notes: 1. For GPIO supported voltages, refer to General Purpose Input and Output (GPIO) on page 92. 2. Maximum overshoot voltage is 2.19 V 3. Min undershoot voltage is -0.28							
Note: For GPIO pads (GP) listed in the Associated Signals below, all functions that are multiplexed on GPIO pads will have the same DC characteristics as the GPIO pads. Refer to General Purpose Input and Output (GPIO) on page 92 for the multiplexed functions on a specific GPIO pad.							
Associated Signals[1]: GP_G00/SD_SDIO_CMD, GP_G01/SD_SDIO_D0, GP_G02/SD_SDIO_D1, GP_G03/SD_SDIO_D2, GP_G04/SD_SDIO_D3, GP_G05/SD_SDIO_CD_N, GP_G06/SD_SDIO_CLK, GP_G07/SD_SDIO_WP							
3.3 V Operation							

continued...

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Input	VIH	Input High Voltage Threshold	0.75 x VCC		V		
	VIL	Input Low Voltage Threshold		0.25 x VCC	V		
	IIL	Input Leakage Current	-14	14	µA		
	CIN	Input Pin Capacitance		14	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.3 * VCC	
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.7 * VCC	
1.8 V Operation							
Input	VIH	Input High Voltage Threshold	0.75 x VCC		V		
	VIL	Input Low Voltage Threshold		0.25 x VCC	V		
	IIL	Input Leakage Current	-14	14	µA		
	CIN	Input Pin Capacitance		14	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.3 * VCC	
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.7 * VCC	
Notes: 1. For GPIO supported voltages, refer to General Purpose Input and Output (GPIO) on page 92 2. If GP_G[00:07] are used for SD functionality, pins are dynamically configured based on SD card capability otherwise if used as GPIO it can be configured to 3.3V or 1.8V.							
Note: For GPIO pads (GP) listed in the Associated Signals below, all functions that are multiplexed on GPIO pads will have the same DC characteristics as the GPIO pads. Refer to General Purpose Input and Output (GPIO) on page 92 for the multiplexed functions on a specific GPIO pad.							
Associated Signals[1]: GP_DSW09/PMC_SLP_WLAN_N, GP_DSW08/PMC_SUSCLK, GP_DSW07, GP_DSW06/PMC_SLP_A_N, GP_DSW05/PMC_SLP_S4_N, GP_DSW04/PMC_SLP_S3_N, GP_DSW03/PMC_PWRBTN_N, GP_DSW02/LAN_WAKE_N, GP_DSW10/PMC_SLP_S5_N, GP_DSW01/PMC_ACPRESENT, GP_DSW00/PMC_BATLOW_N, PMC_SLP_SUS_N, PMC_WAKE_N, PMC_DRAM_RESET_N							
3.3 V Operation							
Input	V_{IH}	Input High Voltage Threshold	0.75 x VCC		V		
	V_{IL}	Input Low Voltage Threshold		0.25 x VCC	V		

continued...

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
	I_{IL}	Input Leakage Current	-10	10	µA		
	C_{IN}	Input Pin Capacitance		14	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	
	R_{pu}	WPU 5K/20K Resistance	1K-50% 5K-70% 20K-35%	1K+100% 5K+70% 20K+35%	Ohm	0.7 * VCC	
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ohm	0.3 * VCC	
Note: For GPIO supported voltages, refer to General Purpose Input and Output (GPIO) on page 92.							
Note: For GPIO pads (GP) listed in the Associated Signals below, all functions that are multiplexed on GPIO pads will have the same DC characteristics as the GPIO pads. Refer to General Purpose Input and Output (GPIO) on page 92 for the multiplexed functions on a specific GPIO pad.							
Associated Signals[1]: GP_B09/PCIE_CLKREQ4_N, GP_B08/PCIE_CLKREQ3_N, GP_B07/PCIE_CLKREQ2_N, GP_B06/PCIE_CLKREQ1_N, GP_B05/PCIE_CLKREQ0_N, GP_B04/CPU_GP_3, GP_B03/CPU_GP_2, GP_B23/DDI2_HPD/TIME_SYNC_0/GSPI1_CS1_N, GP_B22/GSPI1_MOSI, GP_B21/GSPI1_MISO/NFC_CLKREQ, GP_B20/GSPI1_CLK/NFC_CLK, GP_B02/PMC_VRALERT_N, GP_B19/GSPI1_CS0_N, GP_B18/GSPI0_MOSI/UART2A_TXD, GP_B17/GSPI0_MISO/UART2A_RXD, GP_B16/GSPI0_CLK, GP_B15/GSPI0_CS0_N, GP_B14/SPKR_GSPI0_CS1_N, GP_B13/PMC_PLTRST_N, GP_B12/PMC_SLP_S0_N, GP_B11/PMCALERT_N, GP_B10/PCIE_CLKREQ5_N, GP_B01/PMC_CORE_VID1, GP_B00/PMC_CORE_VID0, GP_H09/I2C4_SCL, GP_H08/I2C4_SDA, GP_H07/I2C3_SCL, GP_H06/I2C3_SDA, GP_H05/I2C2_SCL, GP_H04/I2C2_SDA, GP_H03/SX_EXIT_HOLDOFF_N, , GP_H02/MODEM_CLKREQ, GP_H19, GP_H18, GP_H17, GP_H16, GP_H15/AVS_I2S1_SCLK , GP_H14/AVS_I2S2_RXD, GP_H13/AVS_I2S2_TXD/MODEM_CLKREQ, GP_H12/AVS_I2S2_SFRM/CNV_RF_RESET_N, GP_H11/AVS_I2S2_SCLK, GP_H10/CPU_C10_GATE_N, GP_H01/SD_SDIO_PWR_EN_N/CNV_RF_RESET_N, GP_H00, GP_D09/GSPI2_CLK/UART0A_TXD, GP_D08/GSPI2_SPI2_CS0_N/UART0A_RXD, GP_D07, GP_D06, GP_D05, GP_D04, GP_D03/BK_3/SBK_3, GP_D23/I2C5_SCL, GP_D22/I2C5_SDA, GP_D21/CNV_PA_BLANKING, GP_D20/CNV_MFUART2_TXD, GP_D02/BK_2/SBK_2, GP_D19/CNV_MFUART2_RXD, GP_D18/AVS_I2S_MCLK, GP_D17, GP_D16, GP_D15/CNV_WCEN, GP_D14/GSPI2_CS1_N, GP_D13/I2C4B_SCL, GP_D12/I2C4B_SDA, GP_D11/GSPI2_MOSI/UART0A_CTS_N, GP_D10/GSPI2_MISO/UART0A_RTS_N, GP_D01/BK_1/SBK_1, GP_D00/BK_0/SBK_0, GP_C09/UART0_TXD, GP_C08/UART0_RXD, GP_C07/PMC_SUSACK_N, GP_C06/PMC_SUSWARN_N/PMC_SUSPWRDNACK, GP_C05, GP_C04, GP_C03,GP_C23/UART2_CTS_N/CNV_MFUART0_CTS_N, GP_C22/UART2_RTS_N/CNV_MFUART0_RTS_N, GP_C21/UART2_TXD/CNV_MFUART0_TXD, GP_C20/UART2_RXD/CNV_MFUART0_RXD, GP_C02, GP_C19/I2C1_SCL, GP_C18/I2C1_SDA, GP_C17/I2C0_SCL, GP_C16/I2C0_SDA, GP_C15/UART1_CTS_N, GP_C14/UART1_RTS_N, GP_C13/UART1_TXD, GP_C12/UART1_RXD, GP_C11/UART0_CTS_N, GP_C10/UART0_RTS_N, GP_C01, GP_C00							
3.3 V Operation							
Input	V_{IH}	Input High Voltage Threshold	0.75 x VCC		V		
	V_{IL}	Input Low Voltage Threshold		0.25 x VCC	V		
	I_{IL}	Input Leakage Current	-12	12	µA		
	C_{IN}	Input Pin Capacitance		10	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	1K-50% 5K-70% 20K-35%	1K+100% 5K+70% 20K+35%	Ohm	0.7 * VCC	

continued...

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ohm	0.3 * VCC	
1.8 V Operation							
Input	V_{IH}	Input High Voltage Threshold	0.75 x VCC		V		
	V_{IL}	Input Low Voltage Threshold		0.25 x VCC	V		
	I_{IL}	Input Leakage Current	-12	12	µA		
	C_{IN}	Input Pin Capacitance		10	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	1K-50% 5K-70% 20K-35%	1K+100% 5K+70% 20K+35%	Ohm	0.7 * VCC	
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ohm	0.3 * VCC	
Notes: 1. For GPIO supported voltages, refer to General Purpose Input and Output (GPIO) on page 92 2. When eSPI is enabled, SX_EXIT_HOLDOFF_N functionality is not available, and assertion of the signal will not impact Sx exit flows.							
Note: For GPIO pads (GP) listed in the Associated Signals below, all functions that are multiplexed on GPIO pads will have the same DC characteristics as the GPIO pads. Refer to the General Purpose Input and Output (GPIO) on page 92 for the multiplexed functions on a specific GPIO pad.							
Associated Signals[1]: GP_A09/SMB_ALERT_N, GP_A08/SMB_DATA, GP_A07/SMB_CLK, GP_A06/ESPI_RESET_N, GP_A05/ESPI_CLK, GP_A04/ESPI_CS_N, GP_A03/ESPI_IO_3, GP_A02/ESPI_IO_2, GP_A19/PCHHOT_N, GP_A18/USB_OC0_N, GP_A17/DDI0_HPD, GP_A16/DDI1_HPD/TIME_SYNC_1, GP_A15, GP_A14/USB_OC3_N, GP_A13/USB_OC2_N, GP_A12/USB_OC1_N, GP_A11/CPU_GP_1, GP_A10/CPU_GP_0, GP_A01/ESPI_IO_1, GP_A00/ESPI_IO_0, GP_E09/SML_CLK0/SATA_1_GP, GP_E08/SATA_0_GP, GP_E07/SATA_1_DEVSLP, GP_E06/IMGCLKOUT_3, GP_E05/SATA_LED_N, GP_E04/IMGCLKOUT_2, GP_E03/SATA_0_DEVSLP, GP_E23/CNV_RGI_RSP, GP_E22/CNV_RGI_DT, GP_E21/CNV_BRI_RSP, GP_E20/CNV_BRI_DT, GP_E02/IMGCLKOUT_1, GP_E19/IMGCLKOUT_5/PCIE_LNK_DOWN, GP_E18/DDI2_DDC_SDA/BSSB_LS2_TX, GP_E17/DDI2_DDC_SCL/BSSB_LS2_RX, GP_E16/DDI1_DDC_SDA/BSSB_LS1_TX, GP_E15/DDI1_DDC_SCL/BSSB_LS1_RX, GP_E14/DDI0_DDC_SDA/BSSB_LS0_TX, GP_E13/DDI0_DDC_SCL/BSSB_LS0_RX, GP_E12/IMGCLKOUT_4/BSSB_LS3_TX, GP_E11/BSSB_LS3_RX, GP_E10/SML_DATA0, GP_E01, GP_E00/IMGCLKOUT_0							
3.3 V Operation							
Input	V_{IH}	Input High Voltage Threshold	0.75 x VCC		V		
	V_{IL}	Input Low Voltage Threshold		0.25 x VCC	V		
	I_{IL}	Input Leakage Current	-14	14	µA		
	C_{IN}	Input Pin Capacitance		14	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	5K-70%	5K+70%	Ohm	0.7 * VCC	

continued...

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
			20K-25%	20K+35%			
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.3 * VCC	
1.8 V Operation							
Input	V_{IH}	Input High Voltage Threshold	0.75 x VCC		V		
	V_{IL}	Input Low Voltage Threshold		0.25 x VCC	V		
	I_{IL}	Input Leakage Current	-14	14	µA		
	C_{IN}	Input Pin Capacitance		14	pF		
Output	V_{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	IOH=3 mA	Only for 50 ohm mode
	V_{OL}	Output Low Voltage Threshold		0.45	V	IOL=-3 mA	Only for 50 ohm mode
	R_{pu}	WPU 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.3 * VCC	
	R_{pd}	WPD 5K/20K Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ohm	0.7 * VCC	

Notes: 1. For GPIO supported voltages, refer to General Purpose Input and Output (GPIO) on page 92

2. If GP_A[00:06] is used for eSPI it can be configured to 1.8V only otherwise if used as GPIO it can be configured to 3.3V or 1.8V

3. If GP_E[11:18] is used for BSSB it can be configured to 1.8V only otherwise if used as GPIO it can be configured to 3.3V or 1.8V

4. If GP_E[20:23] is used for CNVi it can be configured to 1.8V only otherwise if used as GPIO it can be configured to 3.3V or 1.8V

Table 41. Single-Ended Signal DC Characteristics as Inputs or Outputs

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Associated Signals: INTRUDER_N, PMC_RSMRST_N, PMC_PCH_PWROK, PMC_DSW_PWROK, SRTCRST_N							
Input	VIH	Input High Voltage Threshold	0.65 x VCCRTC	VCCRTC+0.5	V		4, 6
	VIL	Input Low Voltage Threshold	-0.5	0.3 x VCCRTC	V		6
Associated Signals: RTCRST_N							
Input	VIH	Input High Voltage Threshold	0.75 x VCCRTC	V VCCRTC+0.5	V		4, 5, 6
	VIL	Input Low Voltage Threshold	-0.5	0.4 x VCCRTC	V		6
Associated Signals: RTCX1							

continued...

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Input	VIH	Input High Voltage Threshold	0.8	1.2	V		
	VIL	Input Low Voltage Threshold	-0.5	0.1	V		

Notes:
1. The V_{OH} specification does not apply to open-collector or open-drain drivers. Signals of this type must have an external Pull-up resistor, and that is what determines the high-output voltage level.
2. Input characteristics apply when a signal is configured as Input or to signals that are only Inputs. Output characteristics apply when a signal is configured as an Output or to signals that are only Outputs.
3. Vpk-pk minimum for XTAL24 = 500 mV
4. VCCRTC is the voltage applied to the VCCRTC well of the PCH. When the system is in G3 state, it is generally supplied by the coin cell battery. In S5 or greater state, it is supplied by VCCSUS3_3
5. V_{IH} min should not be used as the reference point for T200 timing. Refer T200 specification for the measurement point detail
6. These buffers have input hysteresis. V_{IH} levels are for rising edge transitions and V_{IL} levels are for falling edge transitions.

Table 42. Signal Characteristics

Symbol	Parameter	Minimum	Maximum	Unit	Conditions	Notes
Associated Signals: PCIe*						9, 10
Gen 1						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	0.8	1.2	V		1
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	0.4	1.2	V		
VTX_CM-ACp	TX AC Common Mode Output Voltage (2.5 GT/s)	—	20	mV		
ZTX-DIFF-DC	DC Differential TX Impedance	80	120	Ohm		
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	0.12	1.2	V		1
VRX_CM-ACp	AC peak Common Mode Input Voltage	—	150	mV		
Gen 2						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	0.8	1.2	V		
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	0.4	1.2	V		
ZTX-DIFF-DC	DC Differential TX Impedance	80	120	Ohm		
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	0.12	1.2	V		
VRX_CM-ACp	AC peak Common Mode Input Voltage	—	150	mV		
Gen 3						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	0.8	1.3	V		
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	0.4	1.2	V		
ZTX-DIFF-DC	DC Differential TX Impedance	80	120	Ohm		

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Conditions	Notes
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	Refer to Stressed Voltage Eye Parameters Table in PCIe* Gen 3 industry specifications.				
VRX_CM-ACp	AC peak Common Mode Input Voltage	—	150	mV		
Associated Signals: SATA						
VIMIN-Gen1i	Minimum Input Voltage - 1.5 Gb/s internal SATA	325	—	mVdiff p-p		2
VIMAX-Gen1i	Maximum Input Voltage - 1.5 Gb/s internal SATA	—	600	mVdiff p-p		2
VIMIN-Gen1m	Minimum Input Voltage - 1.5 Gb/s eSATA	240	—	mVdiff p-p		2
VIMAX-Gen1m	Maximum Input Voltage - 1.5 Gb/s eSATA	—	600	mVdiff p-p		2
VIMIN-Gen2i	Minimum Input Voltage - 3.0 Gb/s internal SATA	275	—	mVdiff p-p		2
VIMAX-Gen2i	Maximum Input Voltage - 3.0 Gb/s internal SATA	—	750	mVdiff p-p		2
VIMIN-Gen2m	Minimum Input Voltage - 3.0 Gb/s eSATA	240	—	mVdiff p-p		2
VIMAX-Gen2m	Maximum Input Voltage - 3.0 Gb/s eSATA	—	750	mVdiff p-p		2
VIMIN-Gen3i	Minimum Input Voltage - 6.0 Gb/s internal SATA	240	—	mVdiff p-p		2
VIMAX-Gen3i	Maximum Input Voltage - 6.0 Gb/s internal SATA	—	1000	mVdiff p-p		2
VOMIN-Gen1i,m	Minimum Output Voltage 1.5 Gb/s internal and eSATA	400	—	mVdiff p-p		3
VOMIN-Gen2i,m	Minimum Output Voltage 3.0 Gb/s internal and eSATA	400	—	mVdiff p-p		3
VOMIN-Gen3i	Minimum Output Voltage 6.0 Gb/s internal SATA	200	—	mVdiff p-p		3
VOMAX-Gen3i	Maximum Output Voltage 6.0 Gb/s internal SATA	—	900	mVdiff p-p		3
Associated Signals: USB 2.0						
VDI	Differential Input Sensitivity	0.2	—	V		4, 6
VCM	Differential Common Mode Range	0.8	2.5	V		5, 6
VSE	Single-Ended Receiver Threshold	0.8	2	V		6
VCRS	Output Signal Crossover Voltage	1.3	2	V		6
VOL	Output Low Voltage	—	0.4	V	I_{OL}=5 mA	6
VOH	Output High Voltage	3.3V – 0.5	—	V	I_{OH}=-2 mA	6

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Conditions	Notes
VHSSQ	HS Squelch Detection Threshold	100	150	mV		7
VHSDSC	HS Disconnect Detection Threshold	525	625	mV		7
VHSCM	HS Data Signaling Common Mode Voltage Range	-50	500	mV		7
VHSOI	HS Idle Level	-10	10	mV		7
VHSOH	HS Data Signaling High	360	440	mV		7
VHSOL	HS Data Signaling Low	-10	10	mV		7
VCHIRPJ	Chirp J Level	700	1100	mV		7
VCHIRPK	Chirp K Level	-900	-500	mV		7
New: VDI VCM, VSE, VCRS, VOL, VOH are USB 2.0 FS/LS electrical characteristic.						
Associated Signals: USB 3.2						
VTX-DIFF-PP	Differential Peak to Peak Output Voltage	0.8	1.2	V		
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	0.4	1.2	V		8
VTX_CM-Acp-p	TX AC Common Mode Output Voltage (5GT/s)	—	100	mV		
ZTX-DIFF-DC	DC Differential TX Impedance	72	120	Ohm		
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	0.1	1.2	V		
VRX_CM-ACp	AC peak Common Mode Input Voltage	—	150	mV		

Notes:
1. PCI Express* mVdiff p-p = 2*|PCIE[x]_TXP – PCIE[x]_TXN|; PCI Express* mVdiff p-p = 2*|CIE[x]_RXP – PCIE[x]_RXN|
2. SATA Vdiff, RX (V_{IMAX}/V_{IMIN}) is measured at the SATA connector on the receiver side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]RXP – SATA[x]RXN|.
3. SATA Vdiff, tx (V_{OMIN}/V_{OMAX}) is measured at the SATA connector on the transmit side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]TXP – SATA[x]TXN|
4. V_{DI} = | USBPx[P] – USBPx[N] |
5. Includes VDI range.
6. Applies to Low-Speed/Full-Speed USB.
7. Applies to High-Speed USB 2.0.
8. USB 3.2 mVdiff p-p = 2*|USB3Rp[x] – USB3Rn[x]|; USB 3.1 mVdiff p-p = 2*|USB3Tp[x] – USB3Tn[x]|
9. For PCIe*, GEN1, GEN and GEN3 correspond to the PCIe base specification revision 1, 2 and 3.
10. PCIe* specifications are also applicable to the LAN port.
11. Measurement taken from single-ended waveform on a component test board.
12. Measurement taken from differential waveform on a component test board.
13. VCross is defined as the voltage where Clock = Clock#.
14. Only applies to the differential rising edge (that is, Clock rising and Clock# falling).
15. The maximum voltage including overshoot.
16. The minimum voltage including undershoot.
17. The total variation of all VCross measurements in any particular system. Note that this is a subset of VCross MIN/MAX (VCross absolute) allowed. The intent is to limit VCross induced modulation by setting VCross_Delta to be smaller than VCross absolute.

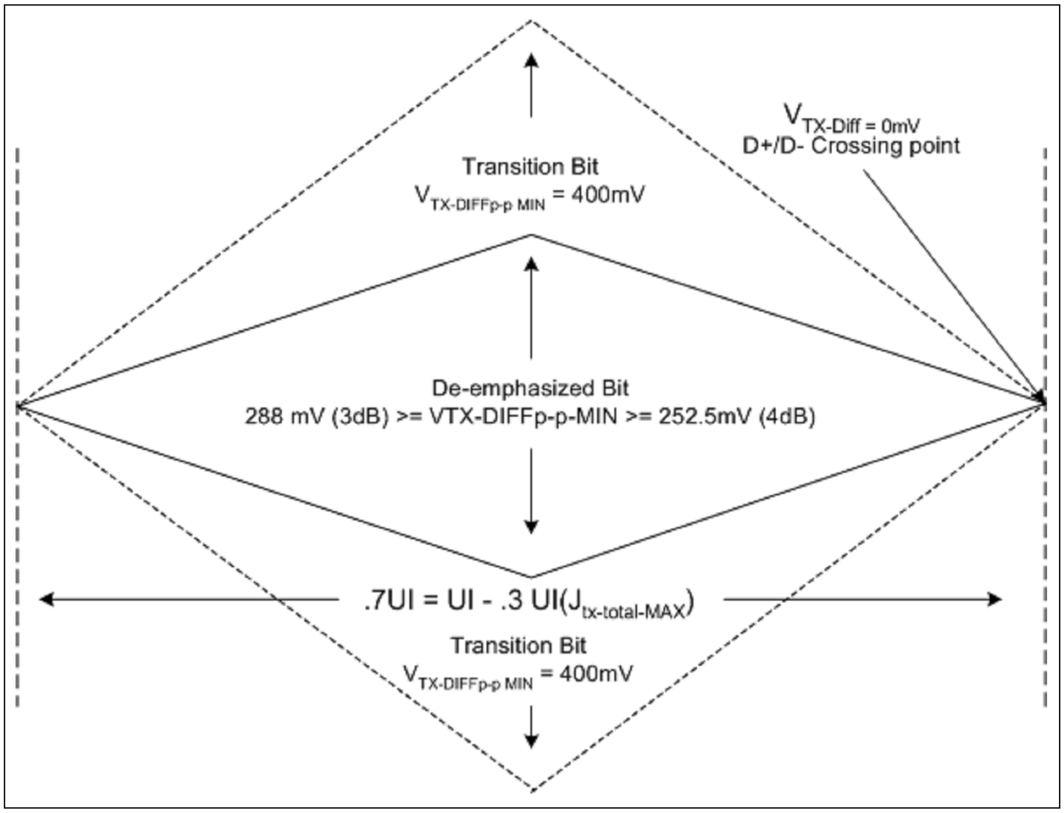
11.3 AC Characteristics

Table 43. PCI Express* Interface Timings

Symbol	Parameter	Minimum	Maximum	Unit	Figures	Notes
Transmitter and Receiver Timings						
UI (Gen1)	Unit Interval – PCI Express	399.88	400.12	ps		5
UI (Gen 2)	Unit Interval – PCI Express	199.9	200.1	ps		5
UI (GEN3)	Unit Interval – PCI Express	124.96	125.03	ps		
TTX-EYE (Gen 1/Gen 2)	Minimum Transmission Eye Width	0.75	—	UI	Figure 11 on page 105	1,2
$T_{\text{TX-EYE-MEDIAN-to-MAX-JITTER}}$ (Gen 1)	Maximum time between the jitter median and maximum deviation from the median	0.125	—	UI		1,2
TRX-EYE (Gen 1)	Minimum Receiver Eye Width	0.4	—	UI	Figure 12 on page 106	3,4
TRX-EYE (Gen 2)	Minimum Receiver Eye Width	0.6	—	UI		3,4

Notes:
1. Specified at the measurement point into a timing and voltage compliance test load and measured over any 250 consecutive TX UIs. (also refer to the Transmitter compliance eye diagram)
2. A $T_{\text{TX-EYE}}$ = 0.70 UI provides for a total sum of deterministic and random jitter budget of $T_{\text{TXJITTER-MAX}}$ = 0.30 UI for the Transmitter collected over any 250 consecutive TX UIs. The $T_{\text{TXEYE-MEDIAN-to-MAX-JITTER}}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total TX jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value.
3. Specified at the measurement point and measured over any 250 consecutive UIs. The test load documented in the PCI Express* specification 2.0 should be used as the RX device when taking measurements (also refer to the Receiver compliance eye diagram). If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as a reference for the eye diagram.
4. A $T_{\text{RX-EYE}}$ = 0.40 UI provides for a total sum of 0.60 UI deterministic and random jitter budget for the Transmitter and interconnect collected any 250 consecutive UIs. The $_{\text{TRX-EYE-MEDIAN-to--MAX-JITTER}}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total 0.6 UI jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value. If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as the reference for the eye diagram.
5. Nominal Unit Interval is 400 ps for 2.5 GT/s and 200 ps for 5 GT/s.

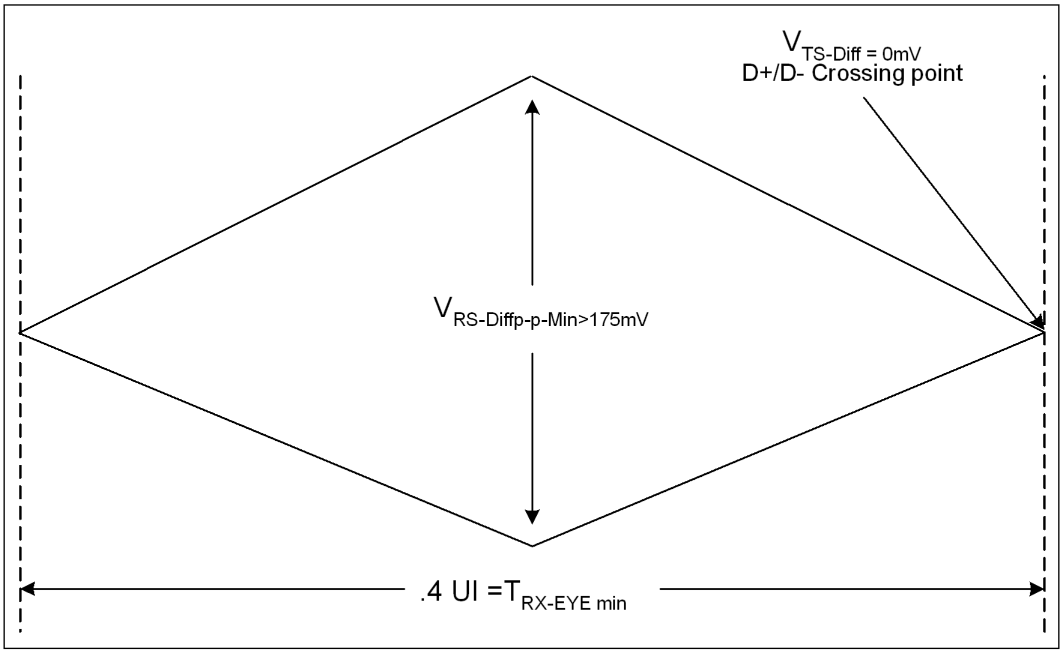
Figure 11. PCI Express* Transmitter Eye


$V_{TX\text{-}Diff}$ = 0mV
D+/D- Crossing point
Transition Bit
$V_{TX\text{-}DIFFp\text{-}p\ MIN}$ = 400mV
De-emphasized Bit
288 mV (3dB) >= VTX-DIFFp-p-MIN >= 252.5mV (4dB)
.7UI = UI - .3 UI($J_{tx\text{-}total\text{-}MAX}$)
Transition Bit
$V_{TX\text{-}DIFFp\text{-}p\ MIN}$ = 400mV

NOTE

Gen1 example is shown for the illustration. Refer to www.pcisig.com for the updated specifications.

Figure 12. PCI Express* Receiver Eye


$V_{TS\text{-}Diff} = 0mV$
D+/D- Crossing point
$V_{RS\text{-}Diffp\text{-}p\text{-}Min>175mV}$
$.4\ UI = T_{RX\text{-}EYE\ min}$

NOTE

Gen1 example is shown for the illustration. Refer to www.pcisig.com for the updated specifications.

Table 44. DDC Characteristics

Signal Group: eDP_VDDEN, eDP_BKLTEN, eDP_BKLTCTL, DDI[0:2]_CTRLCLK, DDI[0:2]_CTRLDATA							
Symbol	Parameter	Standard Mode	Fast Mode		1 MHz		Units
		Maximum	Minimum	Maximum	Minimum	Maximum	
F_{scl}	Operating Frequency	100	0	400	0	1000	kHz
T_r	Rise Time[1]	1000	$20+0.1Cb^2$	300	—	120	ns
T_f	Fall Time[1]	300	$20+0.1Cb^2$	300	—	120	ns

Notes: 1. Measurement Point for Rise and Fall time: V_{IL}(max)–V_{IH}(min)

2. Cb = total capacitance of one bus line in pF. If mixed with High-speed mode devices, faster fall times according to High-Speed mode T_r/T_f are allowed.

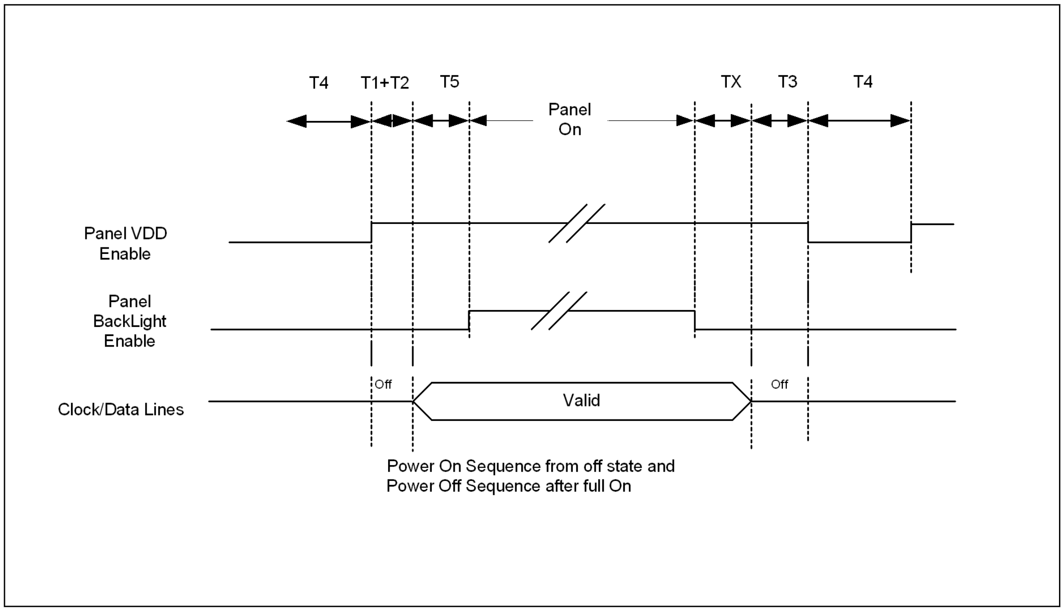
11.3.1 Panel Power Sequencing and Backlight Control

The PCH continues to integrate Panel power sequencing and Backlight control signals for eDP* interfaces on the processor.

This section provides details for the power sequence timing relationship of the panel power, the backlight enable, and the eDP* data timing delivery. To meet the panel power timing specification requirements two signals, eDP_VDDEN and eDP_BKLTEN, are provided to control the timing sequencing function of the panel and the backlight power supplies.

A defined power sequence is recommended when enabling the panel or disabling the panel. The set of timing parameters can vary from panel to panel vendor, provided that they stay within a predefined range of values. The panel VDD power, the backlight on/off state, and the eDP* data lines are all managed by an internal power sequencer.

Figure 13. Panel Power Sequencing



NOTES

1. Support for programming
2. g parameters TX and T1 through T5 using software is provided.

Table 45. DisplayPort* Hot-Plug Detect Interface

Signal Group: DDI[0:2]_HPD						
Symbol	**Parameter**	**Minimum**	**Maximum**	**Unit**	**Figures**	**Notes**
Tir	Input Time Rise	50	500	ps		
Tif	Input Time Fall	50	500	ps		
Tidr	Input Delay Rise	0.3	2.5	ns		
Tidf	Input Delay Fall	0.3	2.5	ns		

Table 46. Clock Timings

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
CLKOUT_PCIE_P/N[4:0]						
Period	Period SSC On	9.849	10.201	ns		Figure 15 on page 113

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
Period	Period SSC Off	9.849	10.151	ns		Figure 15 on page 113
DtyCyc	Duty Cycle	40	60	%		Figure 15 on page 113
V_Swing	Differential Output Swing	300	—	mV		Figure 15 on page 113
Slew_rise	Rising Edge Rate	1.5	4	V/ns		Figure 15 on page 113
Slew_fall	Falling Edge Rate	1.5	4	V/ns		Figure 15 on page 113
	Jitter	—	150	ps	8, 9, 10	
SSC	Spread Spectrum	0	0.5	%	11	
SMBus/SMLink Clock (SMBCLK, SML0CLK)						
fsmb	Operating Frequency	10	100	kHz		Figure 19 on page 119
t18	High Time	4	50	µs	2	
t19	Low Time	4.7	—	µs		Figure 19 on page 119
t20	Rise Time	—	1000	ns		Figure 19 on page 119
t21	Fall Time	—	300	ns		Figure 19 on page 119
SMLink[1,0] (SML0CLK) (Fast Mode: Refer note 15)						
fsmb	Operating Frequency	0	400	kHz		
t18_SMLFM	High Time	0.6	50	µs	2	Figure 19 on page 119
t19_SMLFM	Low Time	1.3	—	µs		Figure 19 on page 119
t20_SMLFM	Rise Time	—	300	ns		Figure 19 on page 119
t21_SMLFM	Fall Time	—	300	ns		Figure 19 on page 119
SMLink[1,0] (SML0CLK) (Fast Mode Plus: Refer note 17)						
fsmb	Operating Frequency	0	1000	kHz		

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
t18_SMLFMP	High Time	0.26	—	µs	2	Figure 19 on page 119
t19_SMLFMP	Low Time	0.5	—	µs		Figure 19 on page 119
t20_SMLFMP	Rise Time	—	120	ns		Figure 19 on page 119
t21_SMLFMP	Fall Time	—	120	ns		Figure 19 on page 119
I^{2}C Clock (Standard Mode)						
fsmb	Operating Frequency	0	100	kHz		
t18_I2CSM	High Time	4	—	µs	2	Figure 19 on page 119
t19_I2CSM	Low Time	4.7	—	µs		Figure 19 on page 119
t20_I2CSM	Rise Time	—	1000	ns		Figure 19 on page 119
t21_I2CSM	Fall Time	—	300	ns		Figure 19 on page 119
I^{2}C Clock (Fast Mode)						
fsmb	Operating Frequency	0	400	kHz		
t18_I2CFM	High Time	0.6	—	µs	2	Figure 19 on page 119
t19_I2CFM	Low Time	1.3	—	µs		Figure 19 on page 119
t20_I2CFM	Rise Time	20	300	ns		Figure 19 on page 119
t21_I2CFM	Fall Time	20 x (V_{DD}/5.5 V)	300	ns		Figure 19 on page 119
I^{2}C Clock (Fast Mode Plus)						
fsmb	Operating Frequency	0	1	MHz		
t18_I2CFMP	High Time	0.26	—	µs	2	Figure 19 on page 119
t19_I2CFMP	Low Time	0.5	—	µs		Figure 19 on page 119

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
t20_I2CFMP	Rise Time	—	120	ns		Figure 19 on page 119
t21_I2CFMP	Fall Time	20 x (V_{DD}/5.5 V)	120	ns		Figure 19 on page 119
I^2C Clock (High Speed Mode, Maximum Bus Capacitance (C_B) = 100 pF)						
fsmb	Operating Frequency	0	3.4	MHz		
t18_I2CHS1	High Time	60	—	ns	2	Figure 19 on page 119
t19_I2CHS1	Low Time	160	—	ns		Figure 19 on page 119
t20_I2CHS1	Rise Time	10	40	ns		Figure 19 on page 119
t21_I2CHS1	Fall Time	10	40	ns		Figure 19 on page 119
I^2C Clock (High Speed Mode, Maximum Bus Capacitance (C_B) = 400 pF)						
fsmb	Operating Frequency	0	1.7	MHz		
t18_I2CHS2	High Time	120	—	ns	2	Figure 19 on page 119
t19_I2CHS2	Low Time	320	—	ns		Figure 19 on page 119
t20_I2CHS2	Rise Time	20	80	ns		Figure 19 on page 119
t21_I2CHS2	Fall Time	20	80	ns		Figure 19 on page 119
HDA_BLK (Intel® High Definition Audio)						
f_{HDA}	Operating Frequency	24	—	MHz		
	Frequency Tolerance	—	100	ppm		
t26a	Input Jitter (refer to Clock Chip Specification)	—	300	ppm		
t27a	High Time (Measured at 0.75 Vcc)	18.75	22.91	ns		Figure 14 on page 112
t28a	Low Time (Measured at 0.35 Vcc)	18.75	22.91	ns		Figure 14 on page 112
Suspend Clock (PMC_SUSCLK)						
fsusclk	Operating Frequency	32		kHz	4	
t39	High Time	9.5	—	µs	4	

continued...

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
t39a	Low Time	9.5	—	µs	4	
XTAL_IN/XTAL_OUT						
ppm[12]	Crystal Tolerance cut accuracy maximum	35 ppm(@ 25 °C ±3 °C)				
ppm[12]	Temp Stability Maximum	30 ppm(10 – 70 °C)				
ppm[12]	Aging Maximum	5 ppm				

Notes:
1. N/A
2. The maximum high time (t18 Max.) provide a simple ensured method for devices to detect bus idle conditions.
3. BCLK Rise and Fall times are measured from 10% VDD and 90% VDD.
4. SUSCLK duty cycle can range from 30% minimum to 70% maximum.
5. Edge rates in a system as measured from 0.8 – 2.0 V.
6. The active frequency can be 5 MHz, 50 MHz, or 62.5 MHz depending on the interface speed. Dynamic changes of the normal operating frequency are not allowed.
7. Testing condition: 1 kOhm Pull-up to Vcc, 1 kOhm Pull-down and 10 pF Pull-down and 1/2 inch trace.
8. Jitter is specified as cycle-to-cycle as measured between two rising edges of the clock being characterized. Period minimum and maximum includes cycle-to-cycle jitter and is also measured between two rising edges of the clock being characterized.
9. On all jitter measurements care should be taken to set the zero crossing voltage (for rising edge) of the clock to be the point where the edge rate is the fastest. Using a Math function = Average(Derivative(Ch1)) and set the averages to 64, place the cursors where the slope is the highest on the rising edge—usually this lower half of the rising edge. The reason this is defined is for users trying to measure in a system it is impossible to get the probe exactly at the end of the Transmission line with large Flip-Chip components. This results in a reflection induced ledge in the middle of the rising edge and will significantly increase measured jitter.
10. Phase jitter requirement: The designated outputs will meet the reference clock jitter requirements from the *PCI Express Base Specification*. The test is to be performed on a component test board under quiet conditions with all clock outputs on. Jitter analysis is performed using a standardized tool provided by the PCI SIG. Measurement methodology is defined in the Intel document "*PCI Express Reference Clock Jitter Measurements*. This is not for ITPXDP_P/N.
11. Spread Spectrum (SSC) is referenced to rising edge of the clock.
12. Total of crystal cut accuracy, frequency variations due to temperature, parasitics, load capacitance variations and aging is recommended to be less than 90 ppm.
13. Spread Spectrum (SSC) is referenced to rising edge of the clock.
14. Spread Spectrum (SSC) of 0.25% on CLKOUT_PCIE[4:0] is used for WiMAX friendly clocking purposes.
15. When SMLink[1,0] is configured to run in Fast Mode (FM) using a soft strap, the supported operating range is 0 Hz ~ 400 kHz, but the typical operating frequency is in the range of 300 kHz – 400 kHz.
16. The 25 MHz output option for CLKOUTFLEX2 is derived from the 25 MHz crystal input to the PCH. The PPM of the 25 MHz output is equivalent to that of the crystal.
17. When SMLink[1,0] is configured to run in Fast Mode Plus (FMP) using a soft strap, the supported operating range is 0 Hz ~ 1 MHz, but the typical operating frequency is in the range of 900 kHz – 1000 kHz. This is the default mode for this interface.
18. Higher fall times are expected at High Speed mode. Validation data shows no functional failures with fall times as low as 9.4 ns and 8.3 ns on SDA and SCL respectively in High Speed mode at 3.3 V with Cb=100 pF.

NOTE

Refer to PCI Local Bus Specification for measurement details.

Figure 14. Clock Timing

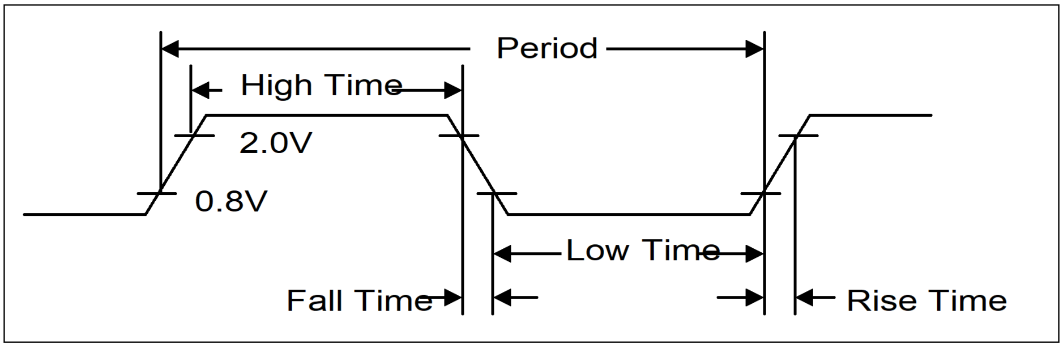


Figure 15. Measurement Points for Differential Waveforms

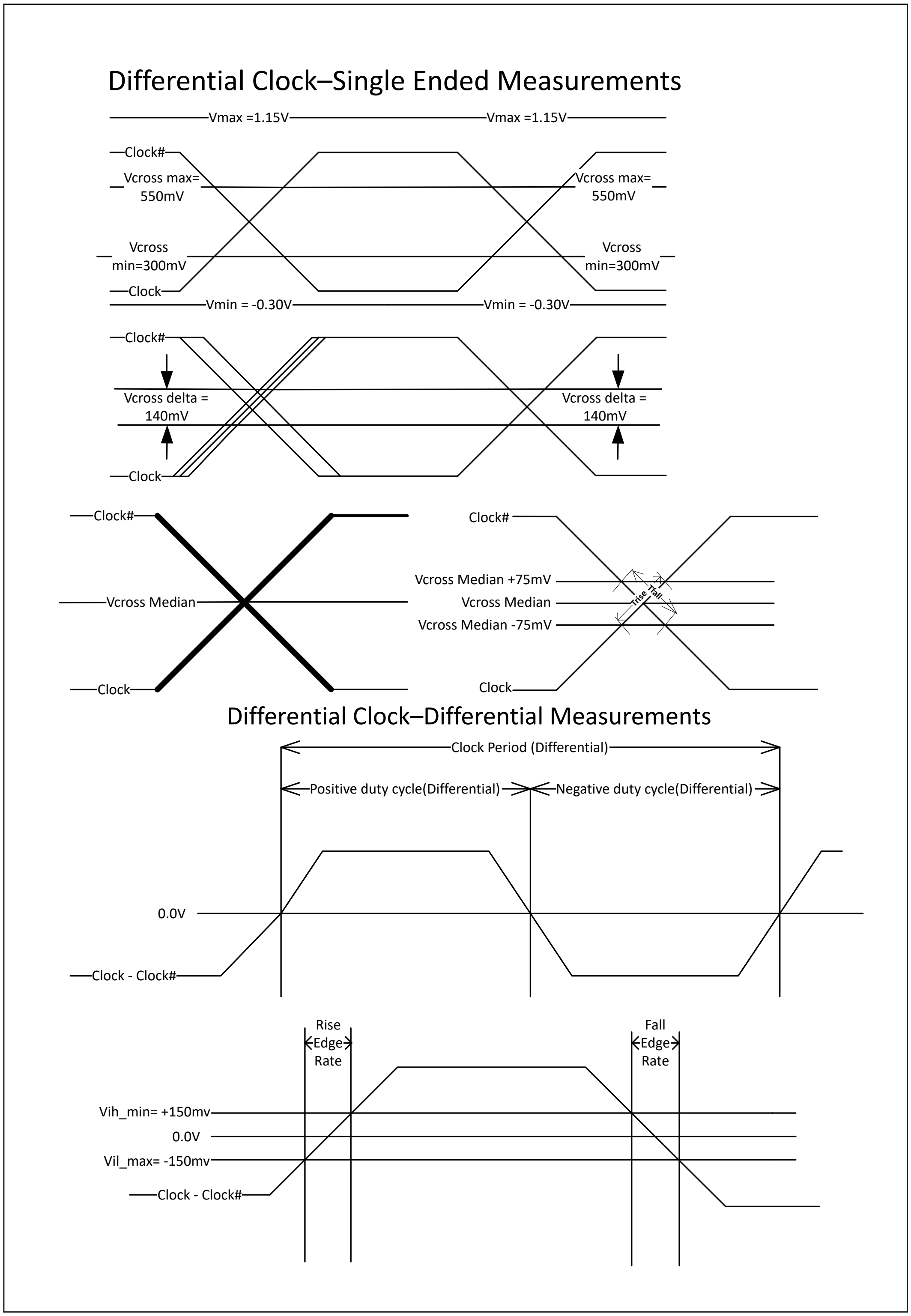


Table 47. USB 2.0 Timing

Sym	Parameter	Minimum	Maximum	Units	Notes	Figure
Full-speed Source (Note 7)						
t100	USBPx+, USBPx- Driver Rise Time	4	20	ns	1,6 CL = 50 pF	Figure 16 on page 115
t101	USBPx+, USBPx- Driver Fall Time	4	20	ns	1,6 CL = 50 pF	Figure 16 on page 115
t102	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	−3.5 −4	3.5 4	ns ns	2, 3	Figure 17 on page 116
t103	Source SE0 interval of EOP	160	175	ns	4	Figure 18 on page 116
t104	Source Jitter for Differential Transition to SE0 Transition	−2	5	ns	5	
t105	Receiver Data Jitter Tolerance - To Next Transition - For Paired Transitions	−18.5 −9	18.5 9	ns ns	3	Figure 17 on page 116
t106	EOP Width: Receiver must accept EOP	82	—	ns	4	Figure 18 on page 116
t107	Width of SE0 interval during differential transition	—	14	ns		
Low-Speed Source (Note 8)						
t108	USBPx+, USBPx − Driver Rise Time	75	300	ns	1,6 CL = 200 pF CL = 600 pF	Figure 16 on page 115
t109	USBPx+, USBPx − Driver Fall Time	75	300	ns	1,6 CL = 200 pF CL = 600 pF	Figure 16 on page 115
t110	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	−25 −14	25 14	ns ns	2,3	Figure 17 on page 116
t111	Source SE0 interval of EOP	1.25	1.5	µs	4	Figure 18 on page 116
t112	Source Jitter for Differential Transition to SE0 Transition	−40	100	ns	5	
t113	Receiver Data Jitter Tolerance - To Next Transition - For Paired Transitions	−152 −200	152 200	ns ns	3	Figure 17 on page 116

continued...

Sym	Parameter	Minimum	Maximum	Units	Notes	Figure
t114	EOP Width: Receiver must accept EOP	670	—	ns	4	Figure 18 on page 116
t115	Width of SE0 interval during differential transition	—	210	ns		

Notes: 1. Driver output resistance under steady state drive is specified at 28 ohms at minimum and 43 ohms at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. Full-speed Data Rate has minimum of 11.97 Mb/s and maximum of 12.03 Mb/s.
8. Low-speed Data Rate has a minimum of 1.48 Mb/s and a maximum of 1.52 Mb/s.

Table 48. USB 3.2 Interface Transmit and Receiver Timings

Sym	Parameter	USB 3.2 Gen 1x1 (5 Gb/s)		USB 3.2 Gen 2x1 (10 Gb/s)		Units
		Minimum	Maximum	Minimum	Maximum	
UI	Unit Interval	199.94	200.06	99.97	100.03	ps
$T_{TX\text{-}EYE}$	Minimum Transmission Eye Width	0.625	—	0.646	—	UI
PU3	Polling Period U3 State	—	100	—	100	mS
PRX-Detect	Polling Period Rx Detect	—	100	—	100	mS

Figure 16. USB Rise and Fall Times

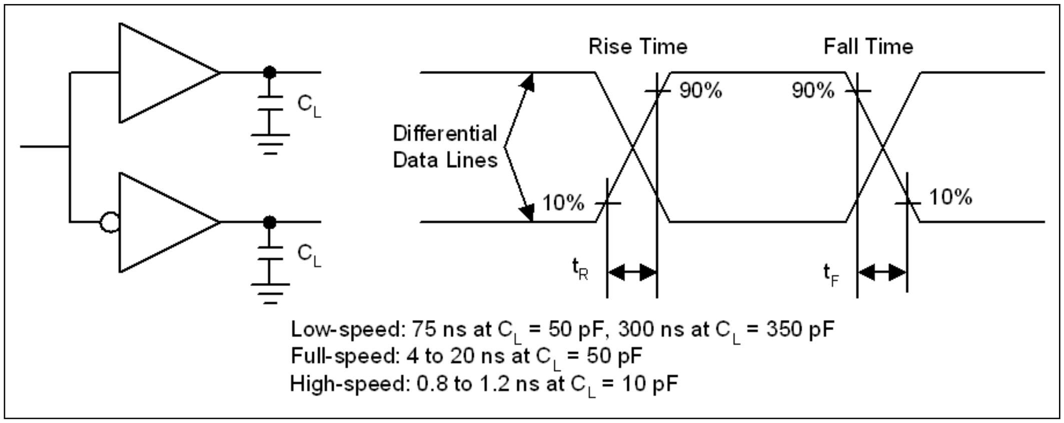

Rise Time
Fall Time
C_L
C_L
Differential Data Lines
90%
90%
10%
10%
t_R
t_F
Low-speed: 75 ns at C_L = 50 pF, 300 ns at C_L = 350 pF
Full-speed: 4 to 20 ns at C_L = 50 pF
High-speed: 0.8 to 1.2 ns at C_L = 10 pF

Figure 17. USB Jitter

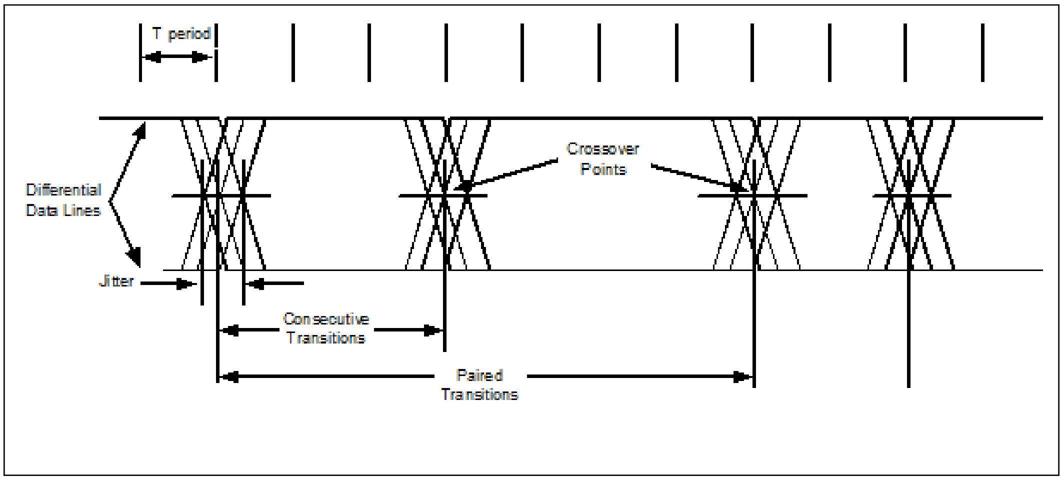


Figure 18. USB EOP Width

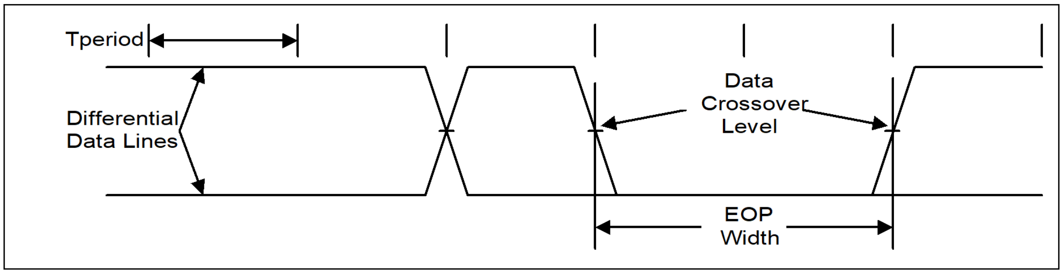


Table 49. SATA Interface Timings

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
UI-3	Gen III Operating Data Period (6 Gb/s)	166.6083	166.6667	ps		
t120gen3	Rise Time	0.2	0.48	UI	1	
t121gen3	Fall Time	0.2	0.48	UI	2	
t122	TX differential skew	—	20	ps		
t123	COMRESET	304	336	ns	3	
t124	COMWAKE transmit spacing	101.3	112	ns	3	
t125	OOB Operating Data period	646.67	686.67	ns	4	

Notes: 1. 20 – 80% at transmitter
2. 80 – 20% at transmitter
3. As measured from 100 mV differential crosspoints of last and first edges of burst
4. Operating data period during Out-Of-Band burst transmissions

Table 50. SMBus Timing

Sym	Parameter	Minimum	Maximum	Units	Notes	Figure
$t130_{100\ kHz}$	Bus Free Time Between Stop and Start Condition	4.7	—	µs		Figure 19 on page 119
$t131_{100\ kHz}$	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4	—	µs		Figure 19 on page 119
$t132_{100\ kHz}$	Repeated Start Condition Setup Time	4.7	—	µs		Figure 19 on page 119
$t133_{100\ kHz}$	Stop Condition Setup Time	4	—	µs		Figure 19 on page 119
$t134_{100\ kHz}$	Data Hold Time	0	—	ns		Figure 19 on page 119
$t135_{100\ kHz}$	Data Setup Time	250	—	ns		Figure 19 on page 119
t136	Device Time Out	25	35	ms	1	
t137	Cumulative Clock Low Extend Time (slave device)	—	25	ms	2	Figure 20 on page 120
t138	Cumulative Clock Low Extend Time (master device)	—	10	ms	3	Figure 20 on page 120
T_{por}	Time in which a device must be operational after power-on reset	—	500	ms		

Notes: 1. A device will timeout when any clock low exceeds this value.

2. t137 is the cumulative time a slave device is allowed to extend the clock cycles in one message from the initial start to stop. If a slave device exceeds this time, it is expected to release both its clock and data lines and reset itself.

3. t138 is the cumulative time a master device is allowed to extend its clock cycles within each byte of a message as defined from start-to-ack, ack-to-ack, or ack-to-stop.

Table 51. I²C and SMLink Timing

Sym[2]	Parameter	Minimum	Maximum	Units	Notes	Figure
$t130_{SM}$	Bus Free Time Between Stop and Start Condition	4.7	—	µs		Figure 19 on page 119
$t130_{FM}$	Bus Free Time Between Stop and Start Condition	1.3	—	µs		Figure 19 on page 119

continued...

Sym[2]	Parameter	Minimum	Maximum	Units	Notes	Figure
$t130_{FMP}$	Bus Free Time Between Stop and Start Condition	0.5	—	µs		Figure 19 on page 119
$t131_{SM}$	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4	—	µs		Figure 19 on page 119
$t131_{FM}$	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.6	—	µs		Figure 19 on page 119
$t131_{FMP}$	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.26	—	µs		Figure 19 on page 119
$t131_{HSM}$	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	160	—	ns		Figure 19 on page 119
$t132_{SM}$	Repeated Start Condition Setup Time	4.7	—	µs		Figure 19 on page 119
$t132_{FM}$	Repeated Start Condition Setup Time	0.6	—	µs		Figure 19 on page 119
$t132_{FMP}$	Repeated Start Condition Setup Time	0.26	—	µs		Figure 19 on page 119
$t132_{HSM}$	Repeated Start Condition Setup Time	160	—	ns		Figure 19 on page 119
$t133_{SM}$	Stop Condition Setup Time	4	—	µs		Figure 19 on page 119
$t133_{FM}$	Stop Condition Setup Time	0.6	—	µs		Figure 19 on page 119
$t133_{FMP}$	Stop Condition Setup Time	0.26	—	µs		Figure 19 on page 119
$t133_{HSM}$	Stop Condition Setup Time	160	—	ns		Figure 19 on page 119
$t134_{SM}$	Data Hold Time	300	—	ns	1	Figure 19 on page 119

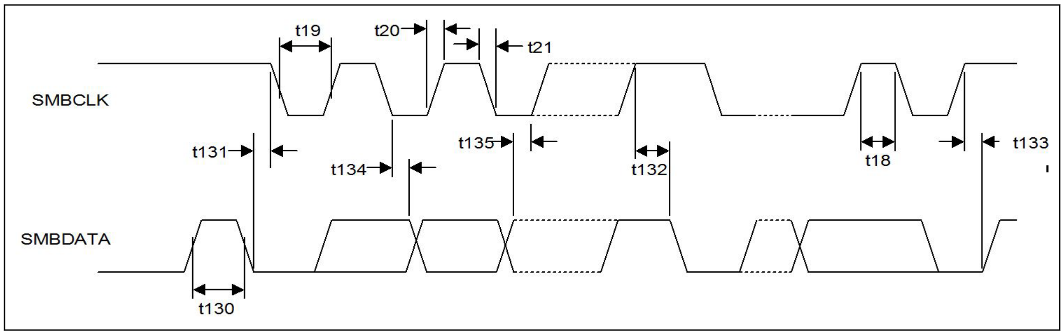
continued...

Sym[2]	Parameter	Minimum	Maximum	Units	Notes	Figure
$t134_{FM}$	Data Hold Time	0	—	ns		Figure 19 on page 119
$t134_{FMP}$	Data Hold Time	0	—	ns		Figure 19 on page 119
$t135_{SM}$	Data Setup Time	250	—	ns		Figure 19 on page 119
$t135_{FM}$	Data Setup Time	100	—	ns		Figure 19 on page 119
$t135_{FMP}$	Data Setup Time	50	—	ns		Figure 19 on page 119
$t135_{HSM}$	Data Setup Time	10	—	ns		Figure 19 on page 119

Notes: 1. t134 has a minimum timing for SMLINK is 300 ns.

2. Timings with the SM designator apply to I2C[0:5] and SMLink[1,0] when operating in Standard Mode, timings with the FM designator apply to I2C[0:5] and SMLink[1:0] when operating in Fast Mode, timings with the FMP designator apply to I2C[0:5] and SMLink[1:0] when operating in Fast Mode Plus and timing with the HSM designator apply only to I2C[0:5] when operating in High Speed Mode.

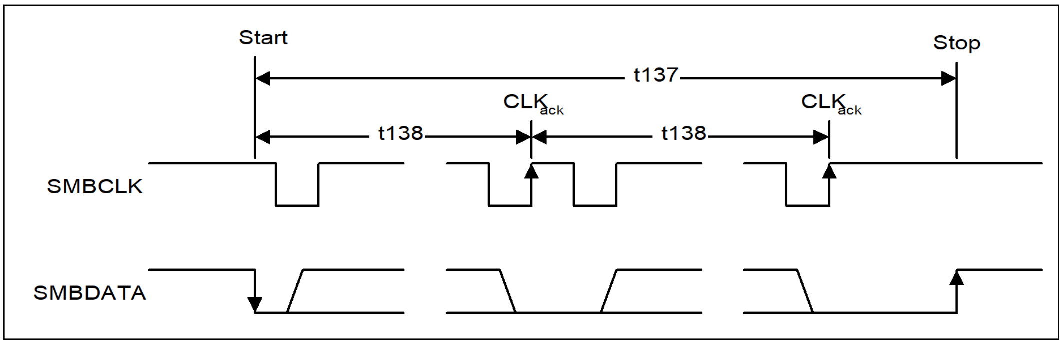
Figure 19. I²C, SMLink and SMBus Transaction



NOTE

txx also refers to txx_SMLFM and txx_SMLFMP, txxx also refers to txxxSMLFM and txxxSMLFMP, SMBCLK also refers to SML0CLK, and SMBDATA also refers to SML[1:0]DATA.

Figure 20. SMBus/SMLink Timeout



NOTE

In this image SMBCLK also refers to SML[1:0]CLK and SMBDATA also refers to SML[1:0]DATA.

Table 52. Intel® High Definition Audio (Intel® HD Audio) Timing

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure

t143	Time duration for which HDA_SDO is valid before HDA_BCLK edge.	6.40	13.20(24 MHz)
			40.00(12 MHz)
				ns		Figure 21 on page 121

t144	Time duration for which HDA_SDO is valid after HDA_BCLK edge.	6.40	13.20(24 MHz)
			40.00(12 MHz)
				ns		Figure 21 on page 121

t145	Setup time for HDA_SDI[1:0] at rising edge of HDA_BCLK	20(24 MHz)
		80(12 MHz)
			—	ns		Figure 21 on page 121

t146	Hold time for HDA_SDI[1:0] at rising edge of HDA_BCLK	3	—	ns		Figure 21 on page 121

Figure 21. Intel® High Definition Audio (Intel® HD Audio) Input and Output Timings

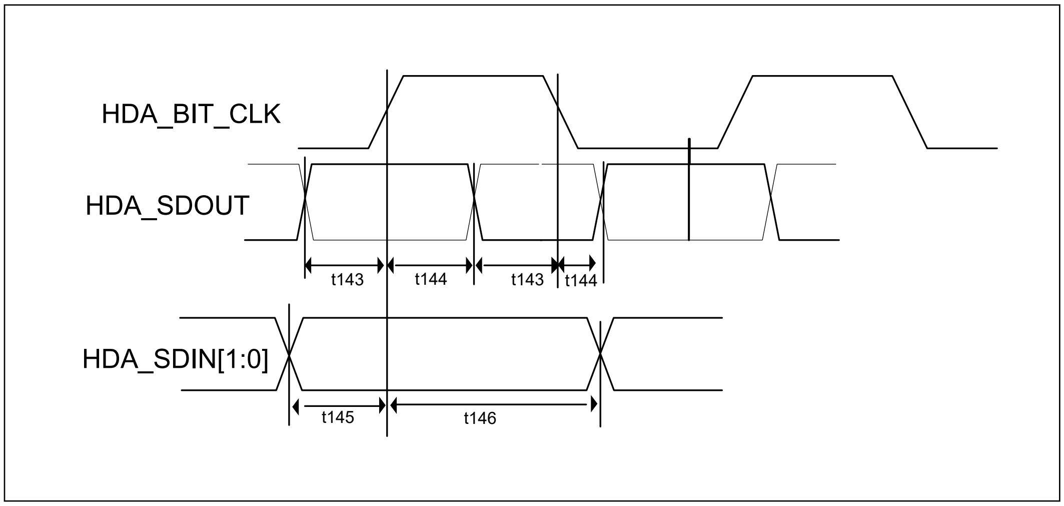


Table 53. DMIC Timing

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
	DMIC_DATA[0:1] Setup Time to DMIC_CLK[0:1] Rising	20	—	ns		Figure 23 on page 122
	DMIC_DATA[0:1] Hold Time from DMIC_CLK[0:1] Rising	1	—	ns		Figure 23 on page 122
Note: DMIC interface rise and fall times are characterized at the PCH package ball.						

Figure 22. Valid Delay from Rising Clock Edge

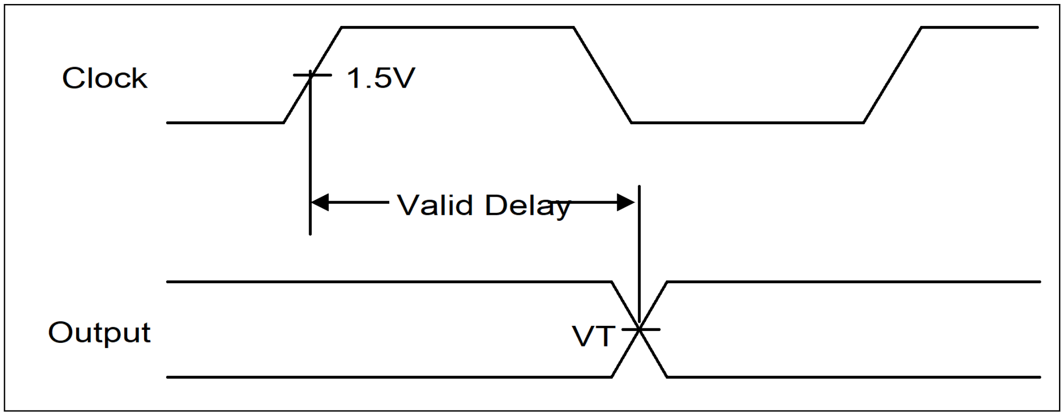


Figure 23. Setup and Hold Times

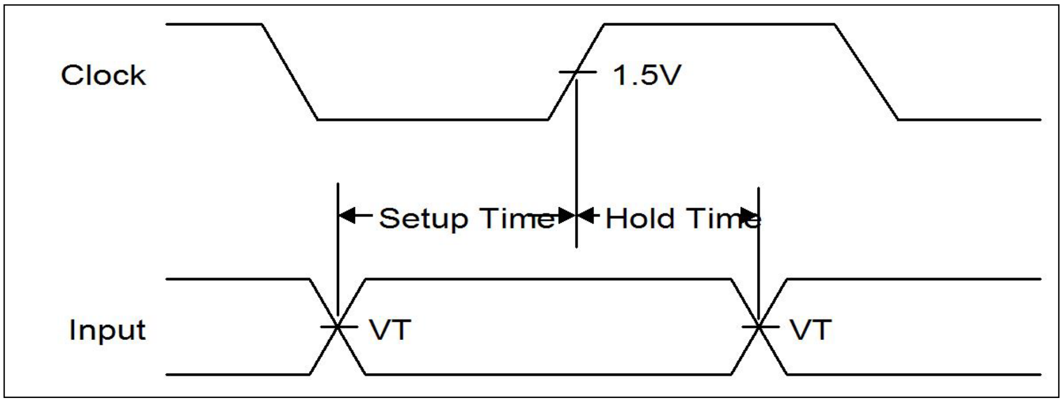


Figure 24. Float Delay

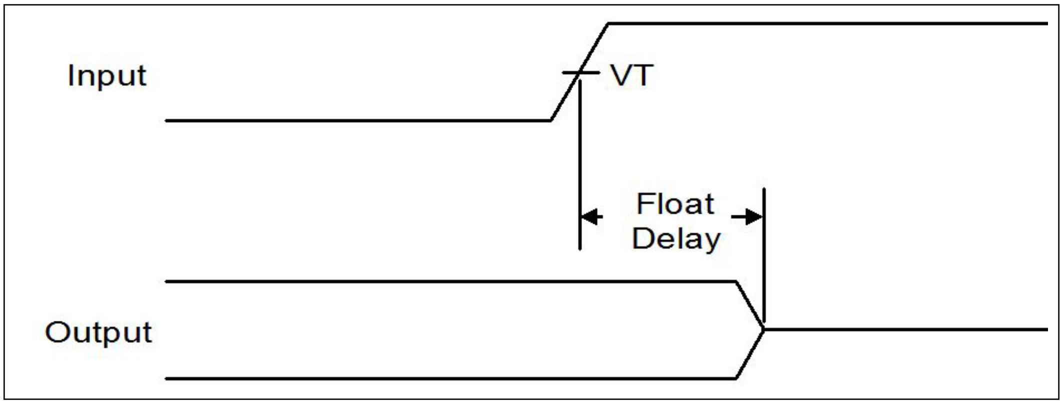


Figure 25. Output Enable Delay

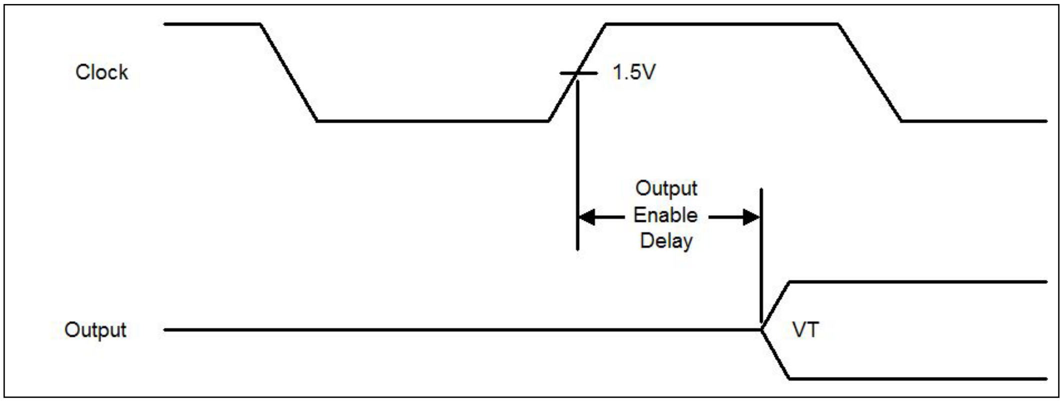


Figure 26. Pulse Width

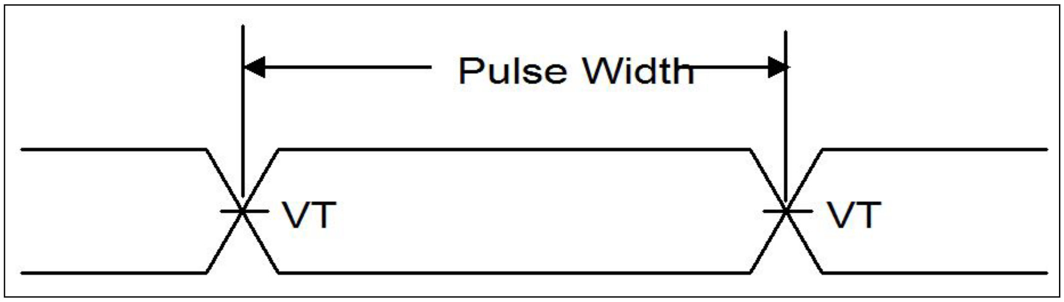


Table 54. Miscellaneous Timings

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t160	SERIRQ Setup Time to PCICLK Rising	7	—	ns		Figure 19 on page 119
t161	SERIRQ Hold Time from PCICLK Rising	0	—	ns		
t162	GPIO, USB Resume Pulse Width	2	—	RTCCLK		Figure 26 on page 123
t163	SPKR Valid Delay from OSC Rising	—	200	ns		Figure 22 on page 121

Table 55. SPI Timings (20 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180a	Serial Clock Frequency	16.8	17.48	MHz	1	
t183a	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-13	14	ns		Figure 28 on page 127
t184a	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	35.0	—	ns		Figure 28 on page 127
t185a	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.50	—	ns		Figure 28 on page 127
t186a	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187a	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t188a	SPI CLK High time	23.84	—	ns		Figure 28 on page 127
t189a	SPI CLK Low time	31.84	—	ns		Figure 28 on page 127

Notes: 1. The typical clock frequency driven by the PCH is 17.14 MHz.
2. Measurement point for low time and high time is taken at 0.5(VCCSPI).
3. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Table 56. SPI Timings (33 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180b	Serial Clock Frequency	29.4	30.6	MHz	1	Figure 28 on page 127
t183b	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-8	8	ns		Figure 28 on page 127
t184b	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	18.0	—	ns		Figure 28 on page 127
t185b	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.50	—	ns		Figure 28 on page 127
t186b	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187b	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127
t188b	SPI CLK High time	16	—	ns		Figure 28 on page 127
t189b	SPI CLK Low time	16	—	ns		Figure 28 on page 127

Notes: 1. The typical clock frequency driven by the PCH is 30 MHz.
2. Measurement point for low time and high time is taken at 0.5(VCCSPI).
3. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Table 57. SPI Timings (50 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180c	Serial Clock Frequency	47.04	48.96	MHz	1	Figure 28 on page 127
t183c	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-3	6.2	ns		Figure 28 on page 127

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t184c	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	7.0	—	ns		Figure 28 on page 127
t185c	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.50	—	ns		Figure 28 on page 127
t186c	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187c	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127
t188c	SPI CLK High time	7.84	—	ns	2, 3	Figure 28 on page 127
t189c	SPI CLK Low time	11.84	—	ns	2, 3	Figure 28 on page 127

Notes: 1. Typical clock frequency driven by the PCH is 48 MHz.
2. When using 48 MHz mode ensure target flash component can meet t188c and t189c specifications. Measurement should be taken at a point as close as possible to the package pin.
3. Measurement point for low time and high time is taken at 0.5(VCCSPI).
4. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Table 58. SPI Timings (60 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180c	Serial Clock Frequency	58.8	61.2	MHz	1	Figure 28 on page 127
t183c	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-3	4.7	ns		Figure 28 on page 127
t184c	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	5	—	ns		Figure 28 on page 127
t185c	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.5	—	ns		Figure 28 on page 127
t186c	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187c	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t188c	SPI CLK High time	5.67	—	ns	2, 3	Figure 28 on page 127
t189c	SPI CLK Low time	10.67	—	ns	2, 3	Figure 28 on page 127

Notes: 1. Typical clock frequency driven by the PCH is 60 MHz.
2. When using 60 MHz mode ensure target flash component can meet t188c and t189c specifications. Measurement should be taken at a point as close as possible to the package pin.
3. Measurement point for low time and high time is taken at 0.5(VCCSPI).
4. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Figure 27. PCH Test Load

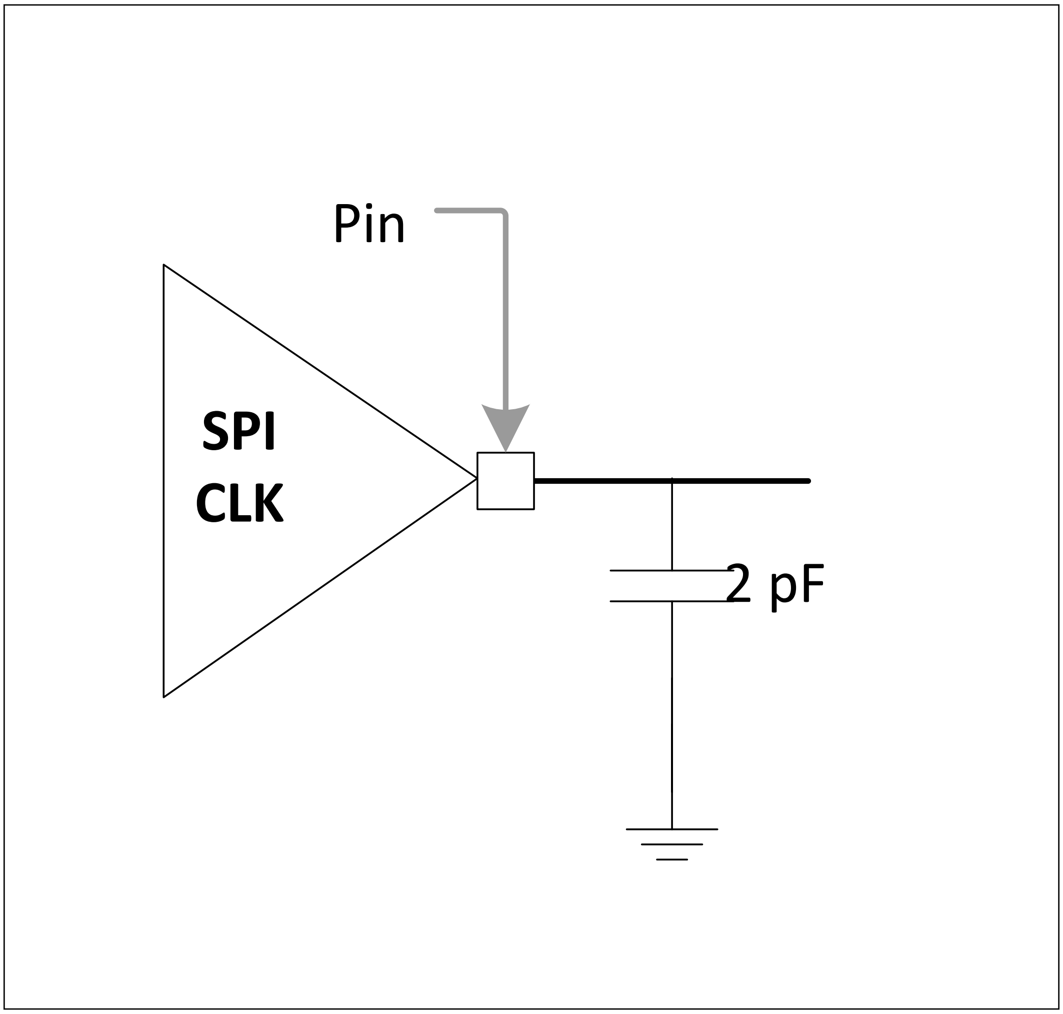


Figure 28. SPI Timings

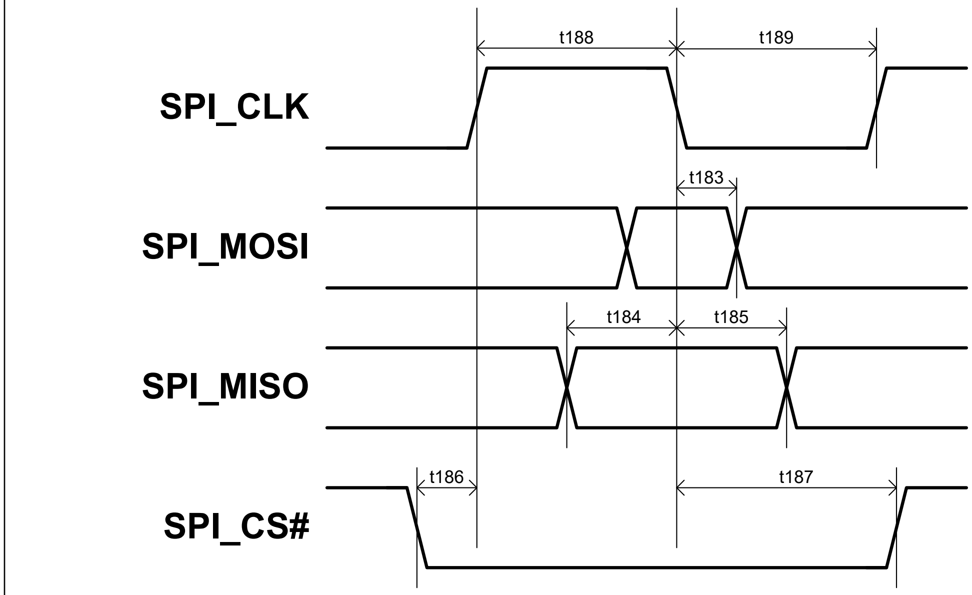


Table 59. GSPI Timings (25 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
F	Serial Clock Frequency	—	25	MHz		Figure 29 on page 128
t183	Tco of SPI MOSI with respect to serial clock falling edge	-15	7.6	ns		Figure 29 on page 128
t184	Setup of SPI MISO and SPI I/O with respect to serial clock rising edge	3.8	—	ns		Figure 29 on page 128
t185	Hold of SPI MISO and SPI I/O with respect to serial clock rising edge	20	—	ns		Figure 29 on page 128
t186	Setup of SPI CS# assertion with respect to serial clock rising edge	20	—	ns		Figure 29 on page 128
t187	Hold of SPI CS# assertion with respect to serial clock falling edge	20	—	ns		Figure 29 on page 128

Figure 29. GSPI Timings

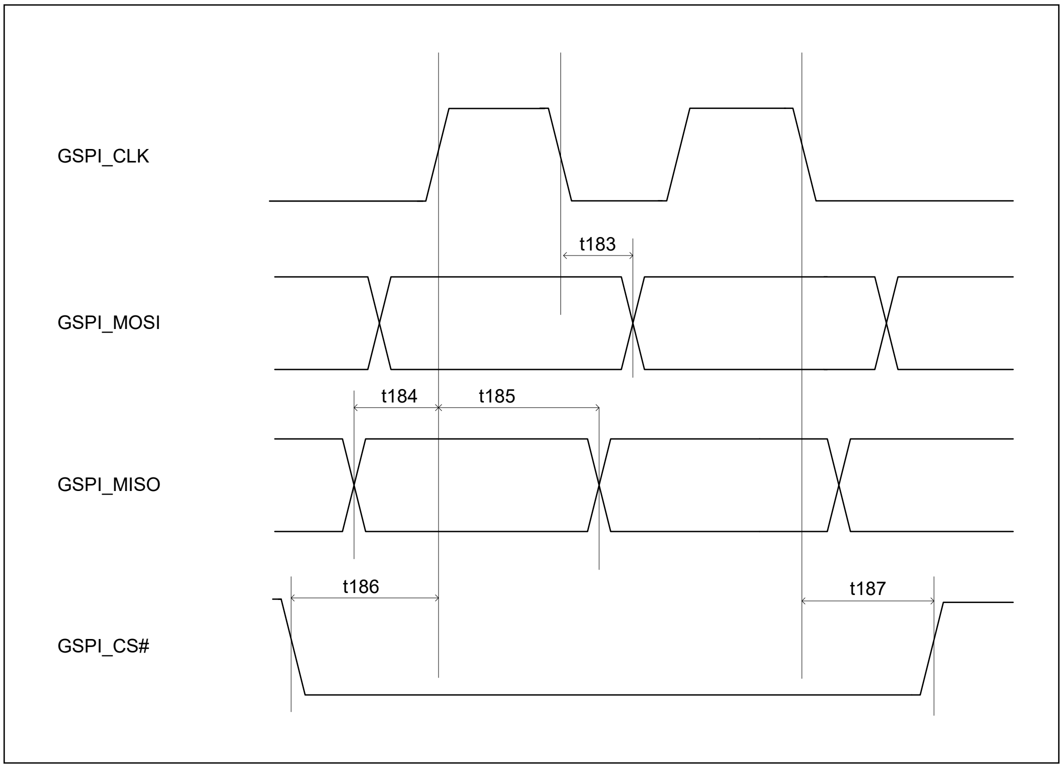


Table 60. UART Timings

Sym	Parameter	Minimum	Maximum	Units	Notes	Figure
F	Operating Frequency	—	6.25	Mbps		
Slew_rise	Output Rise Slope	1.452	2.388	V/ns		
Slew_fall	Output Fall Slope	1.552	2.531	V/ns		

Table 61. I^{2}S Timings - Master Mode

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
SCLK						
FI2S	Clock Frequency	—	12.288	MHz		
FI2S	Clock Frequency	—	9.6	MHz		
SFRM						
TCO	Clock to Output Delay	—8	15	ns		
RXD						
TSU	Setup Time	40	—	ns		
THD	Hold Time	1	—	ns		
TXD						
TCO	Clock to Output Delay	—8	15	ns		

Table 62. I²S Timing - Slave Mode (non S0ix)

Symbol	Parameter	Minimum	Maximum	Units
SCLK				
F_{I2S}	Clock Frequency	—	12.288	MHz
SFRM				
T_{SU}	Setup Time	9	—	ns
T_{HD}	Hold Time	10	—	ns
RXD				
T_{SU}	Setup Time	9	—	ns
T_{HD}	Hold Time	10	—	ns
TXD				
T_{CO}	Clock to Output Delay	0	21	ns

Table 63. I²S Timing - Slave Mode (S0ix)

Symbol	Parameter	Minimum	Maximum	Units
SCLK				
F_{I2S}	Clock Frequency	—	9.6	MHz
SFRM				
T_{SU}	Setup Time	15	—	ns
T_{HD}	Hold Time	10	—	ns
RXD				
T_{SU}	Setup Time	15	—	ns
T_{HD}	Hold Time	10	—	ns
TXD				
T_{CO}	Clock to Output Delay	0	28	ns

Table 64. eSPI Timings (33 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180c	Serial Clock Frequency	29.4	30.6	MHz	1	Figure 28 on page 127
t183c	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-9	8	ns		Figure 28 on page 127
t184c	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	17	—	ns		Figure 28 on page 127
t185c	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.3	—	ns		Figure 28 on page 127

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t186c	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187c	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127
t188c	SPI CLK High time	16	—	ns	2, 3	Figure 28 on page 127
t189c	SPI CLK Low time	16	—	ns	2, 3	Figure 28 on page 127

Notes: 1. Typical clock frequency driven by the PCH is 30 MHz.
2. When using 30 MHz mode ensure target flash component can meet t188c and t189c specifications. Measurement should be taken at a point as close as possible to the package pin.
3. Measurement point for low time and high time is taken at 0.5(VCCSPI).
4. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Table 65. eSPI Timings (66 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180c	Serial Clock Frequency	58.8	61.2	MHz	1	Figure 28 on page 127
t183c	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-3	2.5	ns		Figure 28 on page 127
t184c	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	5.5	—	ns		Figure 28 on page 127
t185c	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.3	—	ns		Figure 28 on page 127
t186c	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		Figure 28 on page 127
t187c	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		Figure 28 on page 127
t188c	SPI CLK High time	8	—	ns	2, 3	Figure 28 on page 127
t189c	SPI CLK Low time	8	—	ns	2, 3	Figure 28 on page 127

Notes: 1. Typical clock frequency driven by the PCH is 60 MHz.
2. When using 60 MHz mode ensure target flash component can meet t188c and t189c specifications. Measurement should be taken at a point as close as possible to the package pin.
3. Measurement point for low time and high time is taken at 0.5(VCCSPI).
4. PCH output timing such as Tco, are simulation values, with a test load of 2pF.

Table 66. eMMC* Signal Group AC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
FCLK	Clock Frequency		200	MHz	typical Value
TX Slew rate	TX pad Slew rate	1.125		V/ns	Test Load @30pF
TCO (HS400 DATA)	Tx Rising/Falling Clock to Data Output Delay (HS400)	-0.65	1.82	ns	1
TCO (HS200 DATA)	Tx Rising Clock to Data Output Delay (HS200)	-0.66	2.356	ns	1
TCO (DDR50 DATA)	Tx Rising/Falling Clock to Data Output Delay (DDR50)	2.5	-	ns	1
TCO (SDR50 DATA)	Tx Rising Clock to Data Output Delay (SDR50)	3	-	ns	1
TCO (DS DATA)	Tx Falling Clock to CMD Output Delay (DS)	-4.673	5.480862	ns	1
TCO (HS400 CMD)	Tx Rising/Falling Clock to CMD Output Delay (HS400)	0.8	-	ns	1
TCO (HS200 CMD)	Tx Rising Clock to CMD Output Delay (HS200)	0.8	-	ns	1
TCO (DDR50 CMD)	Tx Rising/Falling Clock to CMD Output Delay (DDR50)	3	-	ns	1
TCO (SDR50 CMD)	Tx Rising Clock to CMD Output Delay (SDR50)	3	-	ns	1
TCO (DS CMD)	Tx Falling Clock to CMD Output Delay (DS)	3	-	ns	1
TDVW (HS200)	Rx Data Valid Window time to CLK Rising Edge	-1.808		ns	
TSu (DDR50 DATA)	Rx Data Setup Time to CLK Rising/Falling Edge (DDR50)	-3.635		ns	
TH (DDR50 DATA)	Rx Data Hold Time to CLK Rising/Falling Edge (DDR50)	4.318		ns	
TSu (SDR50 DATA)	Rx Data Setup Time to CLK Rising/Falling Edge (SDR50)	-3.751		ns	
TH (SDR50 DATA)	Rx Data Hold Time to CLK Rising/Falling Edge (SDR50)	4.367		ns	
TSu (DS DATA)	Rx Data Setup Time to CLK Rising/Falling Edge (DS)	0.9399		ns	
TH (DS DATA)	Rx Data Hold Time to CLK Rising/Falling Edge (DS)	-4.593		ns	
TWC(DDR50)	CLK Cycle Time (DDR50 Mode)	20		ns	
TWC(SDR50)	CLK Cycle Time (SDR50 Mode)	20		ns	
TWC(DS)	CLK Cycle Time (DS Mode)	40		ns	
TDVW (HS400)	Rx Data Valid wondow to CLK Rising/Falling Edge (HS400)	2.37		ns	
TSu (DDR50 CMD)	Rx CMD Setup Time to CLK Rising/Falling Edge (DDR50)	1.056		ns	

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
TH (DDR50 CMD)	Rx CMD Hold Time to CLK Rising/Falling Edge (DDR50)	2.938		ns	
TSu (SDR50 CMD)	Rx CMD Setup Time to CLK Rising/Falling Edge (SDR50)	-3.751		ns	
TH (SDR50 CMD)	Rx CMD Hold Time to CLK Rising/Falling Edge (SDR50)	4.367		ns	
TSu (DS CMD)	Rx CMD Setup Time to CLK Rising/Falling Edge (DS)	-2.80209		ns	
TH (DS CMD)	Rx CMD Hold Time to CLK Rising/Falling Edge (DS)	-2.366		ns	
TWC (HS200)	CLK Cycle Time (HS200 Mode)	5		ns	
TWC (HS400)	CLK Cycle Time (HS400 Mode)	5		ns	
TWC(DDR50)	CLK Cycle Time (DDR50 Mode)	20		ns	
TWC(SDR50)	CLK Cycle Time (SDR50 Mode)	20		ns	
TWC(DS)	CLK Cycle Time (DS Mode)	40		ns	
Note: 1. SoC output timings are measured at SoC pad with a test load of 2 pF. (50-50%).					

Figure 30. eMMC Output Timing Diagram (High Speed Mode)

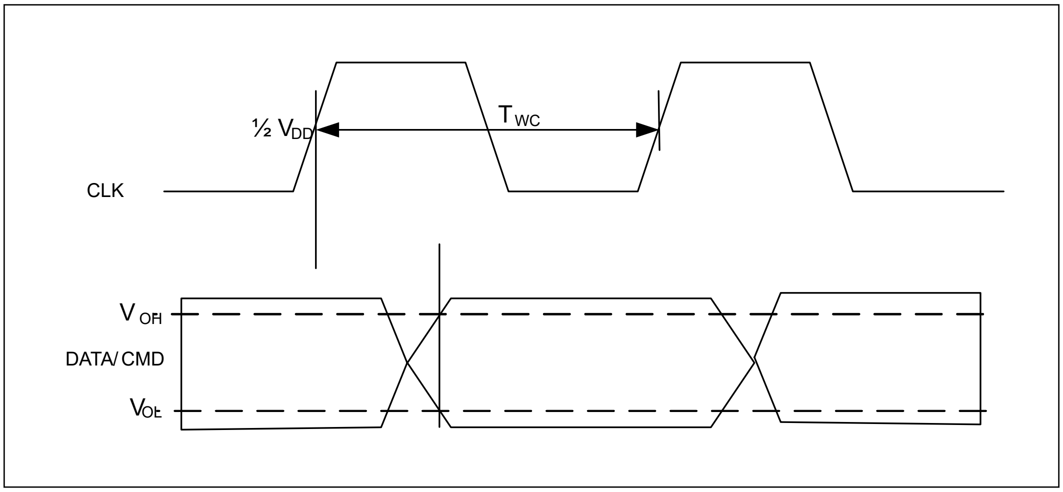


Figure 31. eMMC Timings

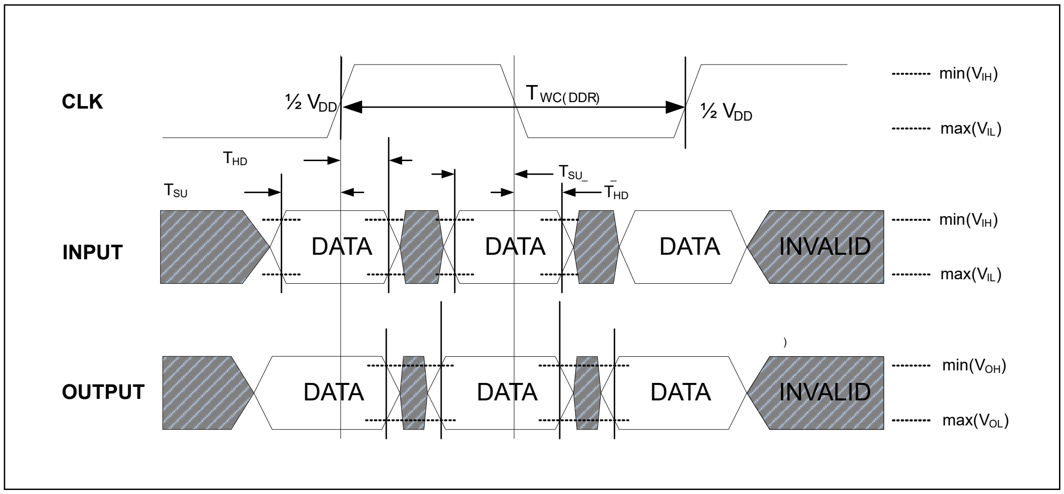


Figure 32. eMMC Input Timing Diagram (High Speed Mode)

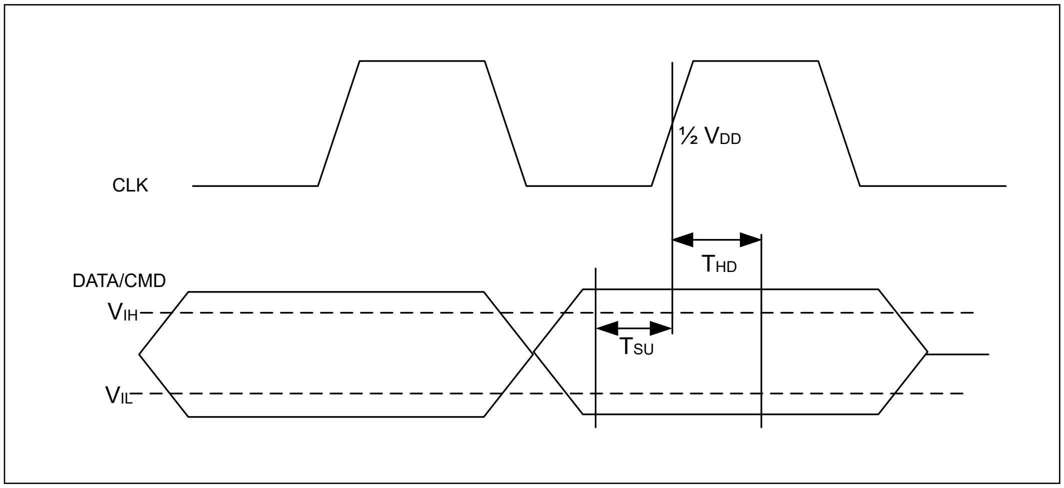
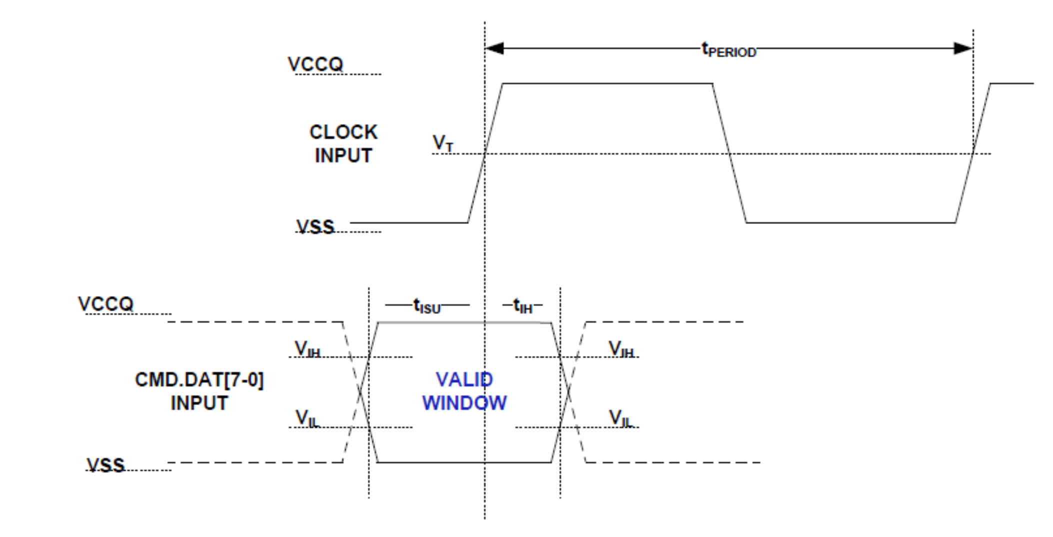


Figure 33. eMMC Input Timing Diagram (HS200/400 Mode)



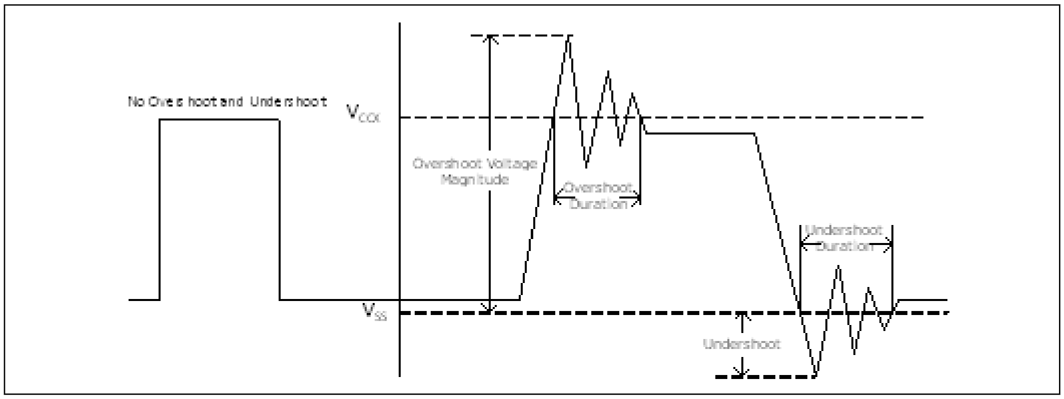
11.4 Overshoot/Undershoot Guidelines

Overshoot (or undershoot) is the absolute value of the maximum voltage above VCC or below VSS. The PCH can be damaged by single and/or repeated overshoot or undershoot events on any input, output, or I/O buffer if the charge is large enough. Baseboard designs that meet signal integrity and timing requirements and that do not exceed the maximum overshoot or undershoot limits listed will ensure reliable I/O performance for the lifetime of the PCH.

Table 67. Overshoot/Undershoot Specifications

Voltage (Vccx)	Overshoot Voltage Magnitude	Overshoot Duration	Undershoot Voltage Magnitude	Undershoot Duration
1.8	1.95	0.6	-0.1	0.6
	1.9	1.2	-0.05	1.2
3.3	3.51	2.5	-0.11	2.5
	3.45	5	-0.05	5

Figure 34. Maximum Acceptable Overshoot/Undershoot Waveform



12.0 CPU Electrical Specifications

This chapter provides information on the following topics:

- DC Specifications

12.1 DC Specifications

NOTES

1. Platform reference voltages are specified at DC only. VCC measurements should be made with respect to the supply voltages specified in Processor Voltage Rails on page 47 and Voltage Rail Electrical Specifications on page 50 for power rail electrical specifications.
2. VIH/OH Max and VIL/OL Minimum values are bounded by VCC and VSS.
3. Care should be taken to read all notes associated with each parameter.

12.1.1 Display Port* Specification

Table 68. Display Port* Channel DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
VIL	Aux Input Low Voltage	-	0.8	V	
VIH	Aux Input High Voltage	2.25	3.6	V	
VOL	Output Low Voltage	-	0.25*VCCIO_ EXT	V	1,2
VOH	Output High Voltage	0.75*VCCIO_ EXT	-	V	1,2
ZTX-DIFF-DC	DC Differential Tx Impedance	100	120	Ω	

Notes: 1. VCCIO_EXT depends on segment.
2. VOL and VOH levels depends on the level chosen by the Platform.

Table 69. Display Port* AUX Channel DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/Figure
VAUX-DIFFp-p	AUX Peak-to-peak Voltage at a transmitting Device	0.29	1.38	V	1
VAUX-_TERM_R	AUX CH termination DC resistance	-	100	ohm	
VAUX-DC-CM	AUX DC Common Mode Voltage	0	2	V	2
VAUX-TURN-CM	AUX turn around common mode voltage	-	0.3	V	3

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes/Figure
IAUX_SHORT	AUX Short Circuit Current Limit	-	90	mA	4
CAUX	AC Coupling Capacitor	75	200	nF	5

Notes:
1. $V_{AUX\text{-}DIFFp\text{-}p}$= 2*|$V_{AUXP}$ – V_{AUXN}|
2. Common mode voltage is equal to V_{bias_Tx} (or V_{bias_Rx}) voltage.
3. Steady-state common mode voltage shift between transmit and receive modes of operation.
4. Total drive current of the transmitter when it is shorted to its ground.
5. All Display Port Main Link lanes as well as AUX CH must be AC coupled.

12.1.2 HDMI* Specifications

Table 70. HDMI* DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
VIL	Aux Input Low Voltage	-	0.8	V	
VIH	Aux Input High Voltage	2.25	3.6	V	
VOL	Output Low Voltage	-	0.25*VCCIO_EXT	V	1,2
VOH	Output High Voltage	0.75*VCCIO_EXT	-	V	1,2
ZTX-DIFF-DC	DC Differential Tx Impedance	100	120	Ω	

Notes:
1. VCCIO_EXT depends on segment.
2. VOL and VOH levels depends on the level chosen by the Platform.

12.1.3 embedded Display Port* Specifications

Table 71. embedded Display Port* DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
V_{OL}	DISP_UTILS Output Low Voltage	—	—	0.1*VCCIO_EXT	V
V_{OH}	DISP_UTILS Output High Voltage	0.9*VCCIO_EXT	—	—	V
R_{UP}	DISP_UTILS Internal pull-up	45	—	—	Ω
R_{DOWN}	DISP_UTILS Internal pull-down	45	—	—	Ω

Table 72. embedded Display Port* AUX Channel DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
VAUX-DIFFp-p	AUX Peak-to-peak Voltage at a transmitting Device	0.29	1.38	V	1
VAUX-_TERM_R	AUX CH termination DC resistance	-	100	Ω	

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
VAUX-DC-CM	AUX DC Common Mode Voltage	0	1.2	V	2
VAUX-TURN-CM	AUX turn around common mode voltage	-	0.3	V	3
IAUX_SHORT	AUX Short Circuit Current Limit	-	90	mA	4
CAUX	AC Coupling Capacitor	75	200	nF	5

Notes: 1. $V_{AUX-DIFFp-p}$= 2*|V_{AUXP} − V_{AUXN}|
2. Common mode voltage is equal to V_{bias_Tx} (or V_{bias_Rx}) voltage.
3. Steady state common mode voltage shift between transmit and receive modes of operation.
4. Total drive current of the transmitter when it is shorted to its ground.
5. All Display Port Main Link lanes as well as AUX CH must be AC coupled.

12.1.4 16550 8-bit Addressing - Debug Driver Compatibility

NOTE

The PCH 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 (byte) 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.

12.1.5 SVID AC Specifications

Table 73. SVID Signal Group AC Specifications

T # Parameter	Minimum	Typ	Maximum	Unit	Notes
VCLK Frequency	10	25	26.25	MHz	1,4
Tco CPU clock to data delay	1.2	-	9.6	ns	-
Tsu_CPU - Setup time of signal VDIO at CPU side	1	-	-	ns	3, 4
Thld_CPU - hold time of signal VDIO at CPU side	3	-	-	ns	3, 4
VCLK Rise Time	0.1	-	5.5	ns	3
VCLK Fall Time	0.1	-	5.5	ns	3
Duty Cycle	40	-	60	%	1,4

Notes:
1. Period and duty cycle are measured with respect to 0.5 * VTT.
2. High and low time is measured with respect to 0.5 * VTT.
3. Rise and Fall times are measured from 0.45 V and 0.65 V.
4. Tperiod, Thigh, Tlow and Duty Cycle variation as a result of internal CPU Clock logic only. Additional variation may be introduced as a result of the Clock MB topology (like different Rpu values or MB impedance).

12.1.6 MIPI* DSI Specification

Table 74. MIPI* DSI DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/ Figure
ILEAK	Pin Leakage current	-650	650	µA	
VCMTX	HS transmit static common-mode voltage	150	250	mV	
|VCMTX(1,0)|	VCMTX mismatch when output is differential-1 or differential-0		5	mV	
|VOD|	HS transmit differential voltage	140	270	mV	
|ΔVOD|	VOD mismatch when output is Differential-1 or Differential-0		14	mV	
VOHHS	HS output high voltage		360	mV	
ZOS	Single-ended output impedance	40	62.5	Ω	
ΔZOS	Single-ended output impedance mismatch		10	%	
VOH	Thevenin output high level	1.1	1.3	V	
VOL	Thevenin output low level	-50	50	mV	
ZOLP	Output impedance of LP transmitter	110		Ω	1
VIH	Logic 1 input voltage	880		mV	
VIL	Logic 0 input voltage, not in ULP state		550	mV	
VHYST	Input hysteresis	25		mV	
VIHCD	Logic 1 Contention threshold	450		mV	
VILCD	Logic 0 Contention threshold		200	mV	
Note: Deviates from MIPI* D-PHY specification Rev 1.1, which has minimum ZOLP of 110 Ω .					

12.1.7 Memory Specifications

Table 75. DDR4 DC Specification

Symbol	Parameter	Minimum	Typical	Maximum	Units	Notes [1]
VIL	Input Low Voltage		0.75*VDDQ	0.68*VDDQ	V	2, 3, 4
VIH	Input High Voltage	0.82*VDDQ	0.75*VDDQ		V	2, 3, 4
IIL	Input Leakage Current (DQ, CK) 0 V 0.2*Vddq 0.8*Vddq	-		1.1	mA	
RON_UP(DQ)	Data Buffer pull-up Resistance	25		60	Ω	5, 12

continued...

Symbol	Parameter	Minimum	Typical	Maximum	Units	Notes [1]
RON_DN(DQ)	Data Buffer pull-down Resistance	26		75	Ω	
RODT(DQ)	On-die termination equivalent resistance for data signals	25		Hi-Z	Ω	6, 12
VODT(DC)	On-die termination DC working point (driver set to receive mode)	0.7 *VDDQ	0.75*VDDQ	0.8*VDDQ	V	12
RON_UP(CK)	Clock Buffer pull-up Resistance	25		60	Ω	5,12
RON_DN(CK)	Clock Buffer pull-down Resistance	25		75	Ω	5,12
RON_UP(CMD)	Command Buffer pull-up Resistance	23		50	Ω	5,12
RON_DN(CMD)	Command Buffer pull-down Resistance	24		57	Ω	5,12
RON_UP(CTL)	Control Buffer pull-up Resistance	23		50	Ω	5,12
RON_DN(CTL)	Control Buffer pull-Down Resistance	24		57	Ω	5,12
RON_UP (DDR_VTT_CTL)	System Memory Power Gate Control Buffer Pull-up Resistance	45		125	Ω	
RON_DN (DDR_VTT_CTL)	System Memory Power Gate Control Buffer Pull- down Resistance	40		130	Ω	
DDR0_VREF_DQ DDR_1_VREF_CA DDR_0_VREF_CA	VREF output voltage	Trainable	VDDQ/2	Trainable	V	
DDR_RCOMP[0]	Command resistance compensation	99	100	101	Ω	8

continued...

Symbol	Parameter	Minimum	Typical	Maximum	Units	Notes [1]
DDR_RCOMP[1]	Data resistance compensation	99	100	101	Ω	8
DDR_RCOMP[2]	ODT resistance compensation	99	100	101	Ω	8

Notes: 1. Unless otherwise noted, all specifications in this table apply to all processor frequencies.Timing specifications only depend on the operating frequency of the memory channel and not the maximum rated frequency.
2. VIL is defined as the maximum voltage level at a receiving agent that will be interpreted as a logical low value.
3. VIH is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value.
4. VIH and VOH may experience excursions above VDDQ.
5. Pull up/down resistance after compensation (assuming ±5% COMP inaccuracy).
6. BIOS power training may change these values significantly based on margin/power trade-off.
7. ODT values after COMP (assuming ±5% inaccuracy). BIOS MRC can reduce ODT strength towards.
8. The minimum and maximum values for these signals are programmable by BIOS to one of the two sets.
9. DDR_RCOMP resistance should be provided on the system board with 1% resistors. SM_RCOMP[x] resistors are to VSS. Values are pre-silicon estimations and are subject to change.
10. PMC_DRAM_RESET_N must have a maximum of 15 ns rise or fall time over VDDQ * 0.30 ±100 mV and the edge must be monotonic.
11. DDR_[1:0]_VREF_CA is defined as VDDQ/2 for DDR4/LPDDR4.
12. RON tolerance is preliminary and might be subject to change.
13. Max-min range is correct but center point is subject to change during MRC boot training.
14. Processor may be damaged if VIH exceeds the maximum voltage for extended periods.

LPDDR4/x Memory Controller DC Specification

Table 76. LPDDR4/x DC Specifications

Symbol	Parameter	Minimum	Typical	Maximum	Units	Notes
VIL	Input Low Voltage		0.2*VDDQ	0.08*VDDQ	V	2, 3, 4
VIH	Input High Voltage	0.35*VDDQ	0.2*VDDQ		V	2, 3, 4
IIL	Input Leakage Current(DQ, CK) 0 V 0.2*VDDQ 0.8*VDDQ	-		1	mA	-
$R_{ON\_UP(DQ)}$	Data Buffer pull-up Resistance	25 (LP4x:23)		60 (LP4x:58)	Ω	5,10
$R_{ON\_DN(DQ)}$	Data Buffer pull-down Resistance	25 (LP4x:26)		72 (LP4x:85)	Ω	5,10
$R_{ODT(DQ)}$	On-die termination equivalent resistance for data signals	28 (LP4x:26)		Hi-Z	Ω	6, 10
$V_{ODT(DC)}$	On-die termination DC working point (driver set to receive mode)	0.15*vddq (LP4x: 0.25* VDDQ)	0.2* VDDQ (LP4x: 0.3* VDDQ)	0.25*VDDQ (LP4x:0.35* VDDQ)	V	10
$R_{ON\_UP(CK)}$	Clock Buffer pull-up Resistance	24 (LP4x:30)		60 (LP4x:59)	Ω	5, 10

continued...

Symbol	Parameter	Minimum	Typical	Maximum	Units	Notes
$R_{ON\_DN(CK)}$	Clock Buffer pull-down Resistance	28		92 (LP4x:94)	Ω	5, 10
$R_{ON\_UP(CMD)}$	Command Buffer pull-up Resistance	26		50	Ω	5, 10
$R_{ON\_DN(CMD)}$	Command Buffer pull-down Resistance	22 (LP4x:20)		67	Ω	5, 10
$R_{ON\_UP(CTL)}$	Control Buffer pull-up Resistance	26		50	Ω	5, 10
$R_{ON\_DN(CTL)}$	Control Buffer pull-down Resistance	22 (LP4x:20)		67	Ω	5, 10
DDR0_VREF_DQ DDR_1_VREF_CA DDR_0_VREF_CA	VREF output voltage	Trainable			V	-
DDR_RCOMP[0]	Command resistance compensation	99	100	101	Ω	8
DDR_RCOMP[1]	Data resistance compensation	99	100	101	Ω	8
DDR_RCOMP[2]	ODT resistance compensation	99	100	101	Ω	8

Notes:
1. Unless otherwise noted, all specifications in this table apply to all processor frequencies. Timing specifications only depend on the operating frequency of the memory channel and not the maximum rated frequency.
2. VIL is defined as the maximum voltage level at a receiving agent that will be interpreted as a logical low value.
3. VIH is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value.
4. VIH and VOH may experience excursions above VDDQ.
5. Pull up/down resistance after compensation (assuming ±5% COMP inaccuracy). Note that BIOS power training may change these values significantly based on margin/power trade-off.
6. ODT values after COMP (assuming ±5% inaccuracy). BIOS MRC can reduce ODT strength towards.
7. The minimum and maximum values for these signals are programmable by BIOS to one of the two sets.
8. LP4_RCOMP resistance should be provided on the system board with 1% resistors. SM_RCOMP[x] resistors are to VSS. Values are pre-silicon estimations and are subject to change.
9. PMC_DRAM_RESET_N must have a maximum of 15 ns rise or fall time over VDDQ * 0.30 ±100 mV and the edge must be monotonic.
10. SM_VREF is defined as VDDQ/2 for DDR4/LPDDR4.
11. RON tolerance is preliminary and might be subject to change.
12. Max-min range is correct but center point is subject to change during MRC boot training.
13. Processor may be damaged if VIH exceeds the maximum voltage for extended periods.

12.1.8 MIPI* CSI Specifications

Table 77. MIPI* CSI DC Specification

Symbol	Parameter	Minimum	Maximum	Units	Notes/Figure
VCMRX(DC)	Common mode voltage HS receive mode	70	330	mV	1,2
VIDTH	Differential input high threshold		70	mV	3
			40	mV	4
VIDTL	Differential input low threshold	-70		mV	3
		-40		mV	4
VIHHS	Single-ended input high impedance		460	mV	1

continued...

Symbol	Parameter	Minimum	Maximum	Units	Notes/Figure
VILHS	Single-ended input low impedance	-40		mV	1
VTERM-EN	Single ended threshold for HS termination enable		450	mV	
ZID	Differential input impedance	80	125	Ω	100 is nominal value

Notes:
1. Excluding possible additional RF interference of 100 mV peak sine wave beyond 450 MHz.
2. This table value includes a ground difference of 50 mV between the transmitter and the receiver, the static common-mode level tolerance and variations below 450 MHz
3. For devices supporting data rates <= 1.5 Gbps.
4. For devices supporting data rates > 1.5 Gbps.

12.1.9 CMOS DC Specifications

Table 78. CMOS Signal Group DC Specifications

Associated Signals: MDSI_DE_TE_1,MDSI_DE_TE_2,PMC_SYS_RESET_N, PROC_PWR_GD, VCCST_PWRGD, SVID_ALERT_N

Symbol	Parameter	Minimum	Maximum	Units	Notes[1]
V_{IL}	Input Low Voltage	—	Vcc*0.3	V	2
V_{IH}	Input High Voltage	Vcc*0.7	—	V	2, 4
R_{ON}	Buffer on Resistance	20	70	Ω	-
I_{LI}	Input Leakage Current	—	±150	μA	3

Notes:
1. Unless otherwise noted, all specifications in this table apply to all processor frequencies.
2. The Vcc referred to in these specifications refers to instantaneous $Vcc_{ST/IO}$.
3. For VIN between "0" V and Vcc_{ST}. Measured when the driver is tri-stated.
4. V_{IH} may experience excursions above Vcc_{ST}.

12.1.10 GTL and Open Drain DC Specification

Table 79. GTL Signal Group and Open Drain Signal DC Specifications

Associated Signals: CPU_JTAG_TRST_N, CPU_JTAG_TMS, CPU_JTAG_TDO, CPU_JTAG_TDI, CPU_JTAG_TCK, CFG[0:15], CFG_AVRB_STB_[0:1]P, CFG_AVRB_STB_[0:1]N, BPM[0:3]_N, JTAG_PREQ_N, JTAG_PRDY_N , CATERR_N , THRMTRIP_N , PROCHOT_N, SVID_CLK, SVID_DATA

Symbol	Parameter	Minimum	Maximum	Units	Notes[1]
V_{IL}	Input Low Voltage (Except CPU_JTAG_TCK, CPU_JTAG_TRST_N)	—	0.6*Vcc	V	2
V_{IH}	Input High Voltage (Except CPU_JTAG_TCK, CPU_JTAG_TRST_N)	0.72*Vcc	—	V	2, 4
V_{IL}	Input Low Voltage (CPU_JTAG_TCK,CPU_JTAG_TRST_N)	—	0.3*Vcc	V	2
V_{IH}	Input High Voltage (CPU_JTAG_TCK,CPU_JTAG_TRST_N)	0.7*Vcc	—	V	2, 4
$V_{HYSTERESIS}$	Hysteresis Voltage	0.2*Vcc	—	V	-
R_{PU}	Pull Up on BPM[0:3] and CFG_AVRB_STB_[0:1]N/P	40	60	Ω	

continued...

Associated Signals: CPU_JTAG_TRST_N, CPU_JTAG_TMS, CPU_JTAG_TDO, CPU_JTAG_TDI, CPU_JTAG_TCK, CFG[0:15], CFG_AVRB_STB_[0:1]P, CFG_AVRB_STB_[0:1]N, BPM[0:3]_N, JTAG_PREQ_N, JTAG_PRDY_N , CATERR_N , THRMTRIP_N , PROCHOT_N, SVID_CLK, SVID_DATA					
Symbol	**Parameter**	**Minimum**	**Maximum**	**Units**	**Notes[1]**
R_{PD}	Pull down on SVID_CLK/DATA, CATERR_N, JTAG_PRDY, JTAG_TDO	7	17	Ω	
R_{PD}	Pull Down on THRMTRIP_N, PROCHOT_N and BPM [0:3]	12	28	Ω	
I_{LI}	Input Leakage Current	—	±150	µA	3

Notes: 1. Unless otherwise noted, all specifications in this table apply to all processor frequencies.
2. The Vcc referred to in these specifications refers to instantaneous $Vcc_{ST/IO}$.
3. For VIN between 0 V and Vcc. Measured when the driver is tri-stated.
4. V_{IH} and V_{OH} may experience excursions above Vcc.
5. JTAG_PRDY_N, JTAG_TDO,CATERR_N, PROCHOT_N,THRMTRIP_N and SVID_CLK are Open Drain
6. Edge rate for BPM[0:3], THRMTRIP_N, CFG_AVRB_STB_[0:1]N/P, SVID_CLK, SVID_DATA, CATERR_N, PROCHOT_N, JTAG PRDY_N and JTAG TDO is between 1.45V/nS to 3.45V/nS.

12.1.11 PECI DC Characteristics

The PECI interface operates at a nominal voltage set by Vcc_{ST}. The set of DC electrical specifications shown in the following table is used with devices normally operating from a Vcc_{ST} interface supply.

Vcc_{ST} nominal levels will vary between processor families. All PECI devices will operate at the Vcc_{ST} level determined by the processor installed in the system.

NOTE

PECI supported frequency range is 3.2 KHz - 1 MHz

Table 80. PECI DC Electrical Limits

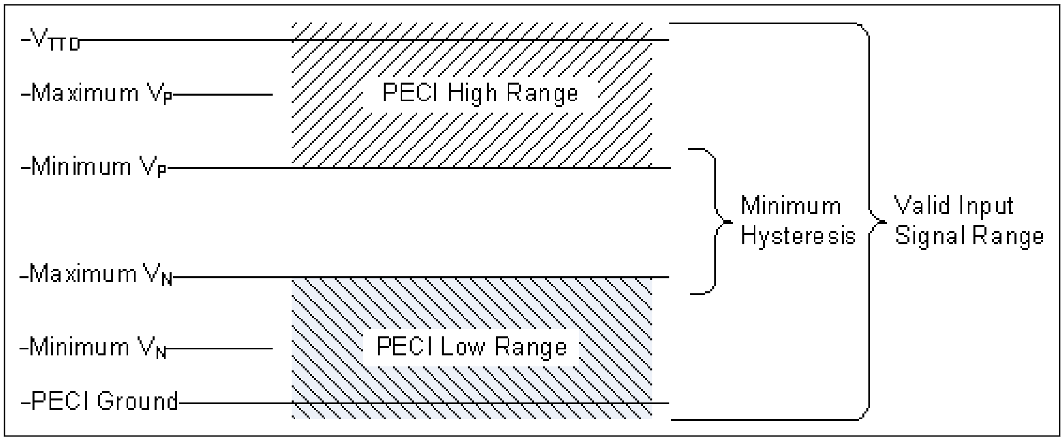
Associated Signal: PECI					
Symbol	**Definition and Conditions**	**Minimum**	**Maximum**	**Units**	**Notes[1]**
R_{up}	Internal pull up resistance	-	39.58	Ω	3
V_{in}	Input Voltage Range	-0.15	Vcc + 0.15	V	-
$V_{hysteresis}$	Hysteresis	0.15 * Vcc	—	V	-
V_{IL}	Input Voltage Low- Edge Threshold Voltage		0.3 * Vcc	V	-
V_{IH}	Input Voltage High- Edge Threshold Voltage	0.7 * Vcc		V	-
C_{bus}	Bus Capacitance per Node	—	10	pF	-
C_{pad}	Pad Capacitance	0.7	1.8	pF	-
Ileak	leakage current	—	50	uA	-

Notes: 1. Vcc_{ST} supplies the PECI interface. PECI behavior does not affect Vcc_{ST} min/max specifications.
2. The leakage specification applies to powered devices on the PECI bus.
3. The PECI buffer internal pull up resistance measured at 0.75* Vcc_{ST}.

Input Device Hysteresis

The input buffers in both client and host models should use a Schmitt-triggered input design for improved noise immunity. Use the following figure as a guide for input buffer design.

Figure 35. Input Device Hysteresis



13.0 Global Device IDs

This chapter provides information on the following topics:

- PCH Global Device IDs
- PCH ACPI IDs
- Compute Global Device ID

13.1 PCH Global Device IDs

Dev ID	Device Function - Device Description	Note
4D80-4D9F	D31:F0 - Enhanced serial peripheral interface Controller(eSPI)	4D87 - Production SKU
4DA0	D31:F1 - Primary to Sideband Bridge(P2SB)	
4DA1	D31:F2 - Power Management Controller(PMC)	
4DA2	RSVD	
4DA3	D31:F4 - System Management Bus (SMBus) Controller	
4DA4	D31:F5 - SPI	Serial Peripheral Interface (SPI) Controller for Flash and TPM
4DA6	D31:F7 - Intel Trace Hub (ITH)	
48A7	RSVD	
4DA8	D30:F0 - LPSS: UART Controller #0	
4DA9	D30:F1 - LPSS: UART Controller #1	
4DAA	D30:F2 - LPSS: SPI Controller #0	
4DAB	D30:F3 - LPSS: SPI Controller #1	
4DB8	D28:F0 - PCI Express Root Port #1	
4DB9	D28:F1 - PCI Express Root Port #2	
4DBA	D28:F2 - PCI Express Root Port #3	
4DBB	D28:F3 - PCI Express Root Port #4	
4DBC	D28:F4 - PCI Express Root Port #5	
4DBD	D28:F5 - PCI Express Root Port #6	
4DBE	D28:F6 - PCI Express Root Port #7	
4DBF	D28:F7 - PCI Express Root Port #8	
4DC4	D26:F0 - SCS	embedded Multi Media Card (eMMC) Controller
4DC5	D25:F0 - LPSS: I2C Controller #4	

continued...

Dev ID	Device Function - Device Description	Note
4DC6	D25:F1 - LPSS: I2C Controller #5	
4DC7	D25:F2 - LPSS: UART Controller #2	
4DC8-4DCF	D31:F3 - cAVS((Audio, Voice, Speech))	
4DD0	D23:F0 - RSVD	Reserved
4DD1	D23:F0 - RSVD	Reserved
4DD2	D23:F0 - SATA	SATA Controller (AHCI)
4DD3	D23:F0 - SATA	SATA Controller (AHCI)
4DD4	D23:F0 - RSVD	Reserved
4DD5	D23:F0 - RSVD	Reserved
4DD6	D23:F0 - SATA	SATA Controller (RAID 0/1/5/10) - Desktop
4DD7	D23:F0 - SATA	SATA Controller (RAID 0/1/5/10) - Mobile
282A	D23:F0 - SATA	SATA Controller (RAID 0/1/5/10) - Mobile
4DD8-4DDD	D23:F0 - RSVD	Reserved
4DDE	D23:F0 - RSVD	SATA Controller (AHCI) Optane-Desktop
4DDF	D23:F0 - RSVD	Reserved
4DE0	D22:F0 - CSE	Host Embedded Controller Interface HECI #1
4DE1	D22:F1 - CSE	HECI #2
4DE4	D22:F4 - CSE	HECI #3
4DE8	D21:F0 - LPSS: I2C Controller #0	
4DE9	D21:F1 - LPSS: I2C Controller #1	
4DEA	D21:F2 - LPSS: I2C Controller #2	
4DEB	D21:F3 - LPSS: I2C Controller #3	
4DED	D20:F0 - USB	USB 3.1 xHCI HC
4DEE	D20:F1 - USB	USB Device Controller (OTG) (xDCI)
4DEF	D20:F2 - PMC	Shared SRAM
4DF0-4DF3	D20:F3 - CNVi	CNVi: WiFi [sku 0:3]
4DF8	D20:F5 - SCS	SCS3:SD Card
4DFA	D18:F5 - SCS	Reserved
4DFB	D18:F6*	LPSS: SPI #2

13.2 PCH ACPI IDs

ACPI ID	Description
INTC34C8	General Purpose Input Output (GPIO) Controller

13.3 Compute Global Device ID

Hex Device ID	Bus	Device	Function	Device
0x4E02	0	0	0	Reserved
0x4E03	0	4	0	Dynamic Platform and Thermal Framework (DPTF)
0x4E04	0	0	0	Reserved
0x4E06	0	0	0	Reserved
0x4E08	0	0	0	Reserved
0x4E0A	0	0	0	Reserved
0x4E0C	0	0	0	Reserved
0x4E0E	0	0	0	Reserved
0x4E10	0	0	0	Reserved
0x4E11	0	8	0	Gaussian Mixture Model and Neural Network Accelerator (GNA)
0x4E12	0	0	0	Processor Transaction Router SKU 4 Core
0x4E14	0	0	0	Processor Transaction Router SKU 2 Core
0x4E16	0	0	0	Reserved
0x4E18	0	0	0	Reserved
0x4E19	0	5	0	IPU
0x4E1A	0	0	0	Reserved
0x4E1C	0	0	0	Reserved
0x4E1E	0	0	0	Reserved
0x4E20		0		Reserved
0x4E22	0	0	0	Processor Transaction Router SKU 2 Core
0x4E24	0	0	0	Processor Transaction Router SKU 4 Core
0x4E26	0	0	0	Processor Transaction Router SKU 4 Core
0x4E28	0	0	0	Processor Transaction Router SKU 4 Core
0x4E29	0	9	0	Intel® Trace Hub
0x4E51	0	2	0	Reserved
0x4E55	0	2	0	GPU 16 EU (Execution Unit)
0x4E61	0	2	0	GPU 24 EU (Execution Unit)
0x4E71	0	2	0	GPU 32 EU (Execution Unit)

14.0 CPU And Device IDs

Table 81. CPUID Format

SKU	CPUID	Reserved [31:28]	Extended Family [27:20]	Extended Model [19:16]	Reserved [15:14]	Processo r Type [13:12]	Family Code [11:8]	Model Number [7:4]	Stepping Value/ID [3:0]
A step	906C0h	Reserved	0000000b	1001	Reserved	00b	0110b	1100b	0000b

- 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 the Intel386™, Intel486™, Pentium, Pentium 4, or 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.
- When EAX is initialized to a value of '1', the CPUID instruction returns the Extended Family, Extended Model, Processor Type, Family Code, Model Number and Stepping ID value in the EAX register. Note that the EDX processor signature value after reset is equivalent to the processor signature output value in the EAX register.

15.0 Audio, Voice, and Speech

- The Converged Audio Voice Speech (cAVS) 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 CPU 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
- The cAVS is fully backward compatible with the Intel HD Audio specification, with the controller implements a number of Output Stream DMA engines and Input Stream DMA engines for data transfers, as well as a Command Output DMA engine and a Response Input DMA engine for control transfers.
- The cAVS also supports I2S audio codecs which are not Intel HD Audio standards. The General Purpose DMA engines has the ability to do simple data transfers or control transfers between system memory and the FIFO in the DSP I/O peripheral interfaces directly, however, these transfers are not optimized for power management.

15.1 Features Supported

This section provides information on the following topics:

- DSP
- Memory
- I/O Peripheral

15.1.1 DSP

The DSP provides a mechanism for intercepting the rendering audio and voice streams (and tones) flowing through the controller’s DMA engines and provides DSP enhancements to the audio. The same controller’s DMA engines may also be used to download DSP function module at run-time, offering flexibility to the Audio DSP processing pipeline creation.

15.1.2 Memory

The central memory block for the cAVS is known as L2 local memory. All the HW based accelerators and DMA engines are able to access certain regions of this central memory as the audio stream buffer. The memory is also used as the working space for the DSP Core, and it can provide processing to the audio stream data flowing through this central memory.

15.1.3 I/O Peripheral

The controller and DSP communicates with the external codec(s) over the audio I/O. These audio I/O connection to codec(s) include the Intel® HD Audio serial link, the Intel® iDisp Audio serial link, or the DSP I/O peripheral for proprietary interfaces (example, I2S). Both the Intel® HD Audio serial link and Intel® iDisp Audio serial link are fully backward compatible with the legacy Intel® HD Audio driver software stack.

15.2 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 OS. Intel® HD Audio controller features are listed as follows:

- Supports data transfers, descriptor fetches, DMA position writes using VC0 or VC1.
- Independent Bus Master logic for 16 general purpose DMA streams: Seven input and Nine output.
- Supports variable length stream slots.
- Supports up to:
  - 16 streams
    - Seven input
      - Two system streams (dedicated)
      - Two system / two offload streams (shared)
      - Two offload streams (dedicated)
      - One feedback stream
  - Nine output
    - Seven system streams
    - Two offload streams
  - 32 bits/sample
  - 192 KHz sample rate
- Supports memory-based command/response transport.
- Three 8-channel universal DMA interfaces for transferring data between memory buffers and peripherals and between memories
- Supports optional Immediate Command/Response mechanism.
- Supports output and input stream synchronization.
- Supports global time synchronization.
- Supports MSI interrupt delivery.
- Support for ACPI D3 and D0 Device States.
- Supports Function Level Reset (FLR)
- Support Converged Platform Power Management (CPPM).
- Support 1 ms of buffering with all DMA running with maximum bandwidth.

- Support 10 ms of buffering with one output DMA and one input DMA running at two channels, 96 KHz, 16 bit audio.

The Input / Output Stream DMA can be individually put into coupled mode where the host and link portion of the DMA will be directed to the associated FIFO and flow-controlled automatically by HW; or put into de-coupled mode where the host and link portion of the DMA will be directed to the unique DSP buffers setup by DSP FW for inserting audio processing pipe stages.

15.2.1 Audio DSP Capabilities

The Audio DSP offload engine is an optional feature providing low power DSP functionality and offload the audio processing operation from host CPU. Audio DSP features are listed as follows:

- Audio DSP with two Tensilica* LX6+HiF3 cores for low power offloaded audio rendering and recording
  - 400 MHz operating frequency in S0
  - 120 MHz operating frequency in S0ix
  - 64KB L1 RAM
  - 1024KB L2 SRAM
- Low power support for Intel® Wake on Voice (Intel® WOV)
- Low power audio playback with post processing
- Low power VoIP and circuit switch voice call with pre-processing
- Various DSP functions optionally provided by DSP Core firware: MP3, AAC, 3rd Party IP Algorithms, etc.

15.3 Direct Attached Digital Microphone (PDM) Interface

The direct attached digital microphone interface is an optional 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:

- Two DMIC PDM interfaces with each interface capable of supporting up to 2 digital MEMs microphones
- Low power always listening support for Intel® Wake on Voice (Intel® WOV)
- Two PCM audio streams (with independent PCM sampling rate: 48 kHz or 16 kHz) per digital mic interface
- Ultrasound reception capable with higher frequency ranges between 3.84 MHz - 4.8 MHz.

15.4 I2S/PCM Interface

The I^{2}S / PCM interface is an optional feature offering connection to the I^{2}S / PCM audio codecs. The I^{2}S / 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^{2}S / 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 link as registers are only addressable through the Audio DSP and its FW. I^2S/PCM Interface features are listed as follows:

- Multiple I2S/PCM ports to support multiple I2S connections
- Can support 3 modes:
  - Slave Mode
  - Slave Mode with Locally Generated Master Clock, or
  - Master Mode
- I^2S audio playback up to 2 ch x 192 kHz x 24 bits
- I^2S audio capture up to 2 ch x 192 kHz x 24 bits
- PCM audio playback up to 8 ch x 48 kHz x 24 bits
- PCM audio capture up to 8 ch x 48 kHz x 24 bits
- Support 3G / 4G modem codec
- Support BT codec HFP / HSP SCO at 8 / 16 kHz
- Support BT codec A2DP at 48 kHz
- Support FM radio codec

15.5 Signal Description

Table 82. Signal Descriptions

Name	Type	Description
Intel® High Definition Audio Signals		
GP_R04/**HDA_RST_N**	O	**Intel® HD Audio Reset**: Master H/W reset to internal/external codecs.
GP_R01/**HDA_SYNC**/ AVS_I2S0_SFRM	O	**Intel® HD Audio Sync:** 48 kHz fixed rate frame sync to the codecs. Also used to encode the stream number.
GP_R00/**HDA_BCLK**/ AVS_I2S0_SCLK	O	**Intel® HD Audio Bit Clock:** Up to 24 MHz serial data clock generated by the Intel® HD Audio controller.
GP_R02/**HDA_SDO**/ AVS_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.
GP_R03/**HDA_SDI0**/ AVS_I2S0_RXD	I	**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.
GP_R05/**HDA_SDI1**/ AVS_I2S1_RXD	I	**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		
GP_R00/HDA_BCLK/ **AVS_I2S0_SCLK**	I/O	**I2S/PCM serial bit clock 0:** Clock used to control the timing of a transfer. Can be generated internally (Master mode) or taken from an external source (Slave mode).

continued...

Name	Type	Description
GP_H15/ **AVS_I2S1_SCLK**	I/O	**I2S/PCM serial bit clock 1:** This clock is used to control the timing of a transfer. Can be generated internally (Master mode) or taken from an external source (Slave mode).
GP_H11/ **AVS_I2S2_SCLK**	I/O	**I2S/PCM serial bit clock 2:** This clock is used to control the timing of a transfer. Can be generated internally (Master mode) or taken from an external source (Slave mode).
GP_R01/HDA_SYNC/ **AVS_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 (Master mode) or taken from an external source (Slave mode).
GP_R06/ **AVS_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 (Master mode) or taken from an external source (Slave mode).
GP_H12/ **AVS_I2S2_SFRM**/ CNV_RF_RESET_N	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 (Master mode) or taken from an external source (Slave mode).
GP_R02/HDA_SDO/ **AVS_I2S0_TXD**	O	**I2S/PCM transmit data (serial data out)0**: This signal transmits serialized data. The sample length is a function of the selected serial data sample size.
GP_R07/**AVS_I2S1_TXD**	O	**I2S/PCM transmit data (serial data out)1**: This signal transmits serialized data. The sample length is a function of the selected serial data sample size.
GP_H13/ **AVS_I2S2_TXD**/ MODEM_CLKREQ	O	**I2S/PCM transmit data (serial data out)2**: This signal transmits serialized data. The sample length is a function of the selected serial data sample size.
GP_R03/HDA_SDI0/ **AVS_I2S0_RXD**	I	**I2S/PCM receive data (serial data in)0:** This signal receives serialized data. The sample length is a function of the selected serial data sample size.
GP_R05/HDA_SDI1/ **AVS_I2S1_RXD**	I	**I2S/PCM receive data (serial data in)1:** This signal receives serialized data. The sample length is a function of the selected serial data sample size.
GP_H14/ **AVS_I2S2_RXD**	I	**I2S/PCM receive data (serial data in)2:** This signal receives serialized data. The sample length is a function of the selected serial data sample size.
GP_D18/ **AVS_I2S_MCLK**	O	**I2S/PCM Master reference clock:** This signal is the master reference clock that connects to an audio codec.
DMIC Interface		
GP_S06/**DMIC_CLK_0**	O	**Digital Mic Clock:** Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz.
GP_S02/**DMIC_CLK_1**	O	**Digital Mic Clock:** Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz.
GP_S07/**DMIC_DATA_0**	I	**Digital Mic Data:** Serial data input from the digital mic.
GP_S03/**DMIC_DATA_1**	I	**Digital Mic Data:** Serial data input from the digital mic.
SoundWire* Interface		
GP_S04/**SNDW1_CLK**	I/O	**SoundWire* Clock:** Serial data clock to external peripheral devices.
GP_S05/**SNDW1_DATA**	I/O	**SoundWire* Data:** Serial data input from external peripheral devices.

15.6 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value (Ohm)
HDA_SYNC	Pull-down	14 K-26 K
HDA_SDO	Pull-down	14 K-26 K
HDA_SDI[1:0]	Pull-down	14 K-26 K
AVS_I2S[2:0]_SFRM	Pull-down	14 K-26 K
AVS_I2S[2:0]_RXD	Pull-down	14 K-26 K
AVS_I2S0_TXD	Pull-down	14 K-26 K
AVS_I2S[2:0]_SCLK	Pull-down	14 K-26 K
DMIC_DATA[1:0]	Pull-down	14 K-26 K
SNDW1_DATA	Pull-down	5 K
SPKR	Pull-down	14 K-26 K

15.7 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S3/S4/S5	Deep Sx
High Definition Audio Interface					
HDA_RST_N	Primary	Driven Low	Driven Low	Driven Low	OFF
HDA_SYNC	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
HDA_BLK	Primary	Driven Low	Driven Low	Driven Low	OFF
HDA_SDO	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
HDA_SDI[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
I2S/PCM Interface					
AVS_I2S[2:1]_SCLK	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
AVS_I2S[2:0]_SFRM	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
AVS_I2S0_TXD	Primary	Internal Pull-down	Driven Low	Low then disabled (Refer Note)	OFF
AVS_I2S[2:1]_TXD	Primary	Driven Low	Driven Low	Driven Low	OFF
AVS_I2S[2:0]_RXD	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
AVS_I2S_MCLK	Primary	Driven Low	Driven Low	Driven Low	OFF
DMIC Interface					
DMIC_CLK[1:0]	Primary	Driven Low	Driven Low	Driven Low	OFF
DMIC_DATA[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF

continued...

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S3/S4/S5	Deep Sx
SoundWire* Interface					
SNDW1_DATA	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SNDW_CLK1	Primary	Driven Low	Driven Low	Driven Low	OFF

Notes: 1. I2S0_TXD are straps in which the pull-down only occurs during the sampling window and then the pull-ups are disabled.
2. Reset reference for primary well pins is PMC_RSMRST_N.

15.8 References

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

16.0 Connectivity Integrated (CNVi)

Connectivity Integrated (CNVi) is a general term referring to a family of connectivity solutions.

The Integrated Connectivity (CNVi) solution consists of the following entities:

- The PCH which contains the Connectivity Controller IP.
- Companion RF chip that is in a pre-certified module.
- The CNVio signals electrical characteristics are similar to the MIPI-DPHY standard.

Table 83. References

Specification	Location
M.2 Specification	https://pcisig.com/specifications/pciexpress/M.2_Specification/
MIPI® Alliance specification for D-PHY v2.0	http://www.mipi.org/specifications/

16.1 Features Supported

- The Processor adopts CNVio Gen2, has two lanes in each direction, which supports the following bit rates: (per lane)
- 2.5 GBit/Sec Rate with 1250MHz clock rate in Gen2 Mode, however.
- When pair with CNVio Gen1 based module the clock rate is 660 MHz.
- 802.11ax [WIFI 6]/ac
- BT 5.0 / BTLE Support

16.2 Signal Description

Name	Type	Description
GPIO Fixed Function		
GP_H11/AVS_I2S2_SCLK	I/O	For CNVi: Unused For standard CNV with UART host support: Optional Bluetooth I^{2}S bus clock
GP_H12/AVS_I2S2_SFRM/ **CNV_RF_RESET_N**	I/O	For CNVi: RF companion (CRF) reset signal, active low. Require a 75KOhm Pull-Down on platform/motherboard level. Recommended not use it for bootstrapping during early Platform init flows. For standard CNV with UART host support: Optional Bluetooth I^{2}S bus sync
GP_H13/AVS_I2S2_TXD/ **MODEM_CLKREQ**	O	For CNVi: Clock request signal. Used to request the RF companion clock (38.4M Ref clock); In PCH this function is not used, BUT this signal is also used for CNVi Init flow, so it must be connected on platform level even when clk sharing ability is not used/feasible. PCH is using internal clk (38.4 MHz clk) and **NOT taking this clk from the CRF** (as was optional in previous generations) For standard CNV with UART host Bluetooth* support: Optional Bluetooth* I^{2}S bus data output (input to BT module).

continued...

Name	Type	Description
GP_H14/AVS_I2S2_RXD	I	For CNVi: Unused. For standard CNV with UART host support: Optional Bluetooth* I^{2}S bus data output (input to BT module)
GP_E20/**CNV_BRI_DT**	O	For CNVi: BRI bus TX. For standard CNV CNV with UART host support: BT UART RTS# *Note:* Require a 100-50-20KOhm (any of) Pull-up on platform/motherboard level. Recommended not use it for bootstrapping during early Platform init flows
GP_E21/**CNV_BRI_RSP**	I	For CNVi: BRI bus RX. For standard CNV CNV with UART host support: BT UART RXD
GP_E22/**CNV_RGI_DT**	O	For CNVi: RGI bus TX. For standard CNV with UART host support: BT UART TXD
GP_E23/**CNV_RGI_RSP**	I	For CNVi: RGI bus RX. For standard CNV with UART host support: BT UART CTS#
GP_H01/ SD_SDIO_PWR_EN_N/ **CNV_RF_RESET_N**	O	For CNVi (main): RF companion (CRF) reset signal, active low. Require a 75KOhm Pull-Down on platform/motherboard level. Recommended not use it for bootstrapping during early Platform Init flows.
GP_H02/**MODEM_CLKREQ**	O	For CNVi(main): Clock request signal. Used to request the RF companion clock (38.4M Ref clock); In PCH this function is not used, BUT this signal is also used for CNVi Init flow, so it must be connected on platform level even when clk sharing ability is not used/feasible. PCH using internal clk (38.4 MHz clk) and NOT taking this clk from the CRF (as was optional in previous generations)
GP_D21/ **CNV_PA_BLANKING**	I	For CNVi and standard CNV: Optional WLAN/BT-WWAN coexistence signal COEX3. Used to be co-existence signal for external GNSS solution
GP_D19/ **CNV_MFUART2_RXD**	I/O	For CNVi and standard CNV: Optional WLAN/BT-WWAN coexistence signal COEX (Input)
GP_D20/ **CNV_MFUART2_TXD**	I/O	For CNVi and standard CNV: Optional WLAN/BT-WWAN coexistence signal COEX (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-
CNV_WT_RCOMP	O	WiFi DPHY RCOMP, analog connection point for an external bias resistor to ground
Selectable Special Purpose I/O		

continued...

Name	Type	Description
USB2P_8		Bluetooth USB host bus (positive) for standard CNV. Optional to connect to a Bluetooth USB+ pin on the Bluetooth module. Port 8 is the recommended port .
USB2N_8		Bluetooth USB host bus (negative) for standard CNV. Optional to connect to a Bluetooth USB- pin on the Bluetooth module. Port 8 is the recommended port.
W_disable1#(GPIO)	I	Used for Wi-Fi* RF-Kill control. This pin can be connected to a platform switch or to SoC GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init).
W_disable2#(GPIO)	I	Used for BT RF-Kill control. This pin can be connected to a platform switch or to SoC GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init).

16.3 Integrated Pull-ups and Pull-downs

Signal	Resistor	Value	Notes
CNV_BRI_RSP	Pull up	~120 kohm	In RF Companion Chip
CNV_RGI_RSP	Pull up	~120 kohm	In RF Companion Chip

16.4 Platform PU/PD Requirements

I/F	Signals	PU/PD in Platform	Comments
BRI/RGI Bluetooth* UART	CNV_RGI_DT	PU (20kohm)	This pull is required so that the SOC will be able to reliably detect that the CRF is present at power-up. However, it is possible to increase the resistor to 50K or even to to 100K instead of 20K.
Init signals	RF_RESET_N	PD (75kohm)	It is highly encouraged to increase this resistor (or allow to switch it off when CNVi is active; not sure this is possible at the platform level). This resistor consumes power (43uW) all the time. This recommendation applicable only for reset muxed with GP_F04
A4WP indication	A4WP_PRESENT	PD (75kohm)	Native function A4WP is not supported. The pin can instead be used as GPIO (when BIOS programs the pin to GPIO functionality). It is recommended to have an external pull down on the pin regardless of the pin being used or not to minimize power consumption. If the pin is used as GPIO, there should NOT be any on-board device driving the pin high until BIOS programs it to GPIO functionality.

16.5 I/O Signal Planes and States

Signal Name	Power plane	During Reset[1]	Immediately After Reset[1]	S3/S4/S5	Deep Sx
CNV_RF_RESET_N	Primary	Driven	Driven	Driven	OFF
MODEM_CLKREQ	Primary	Driven	Driven	Driven	OFF

continued...

Signal Name	Power plane	During Reset[1]	Immediately After Reset[1]	S3/S4/S5	Deep Sx
CNV_PA_BLANKING	Primary	Undriven	Undriven	Undriven	OFF
CNV_MFUART2_RXD	Primary	Undriven	Undriven	Undriven	OFF
CNV_MFUART2_TXD	Primary	Undriven	Undriven	Undriven	OFF
CNV_BRI_DT	Primary	Driven	Driven	Driven	OFF
CNV_BRI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)	OFF
CNV_RGI_DT	Primary	Driven	Driven	Driven	OFF
CNV_RGI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)	OFF
CNV_WT_CLKP	Primary	Undriven	Undriven	Driven	OFF
CNV_WT_CLKN	Primary	Undriven	Undriven	Driven	OFF
CNV_WT_D0P	Primary	Undriven	Undriven	Driven	OFF
CNV_WT_D0N	Primary	Undriven	Undriven	Driven	OFF
CNV_WT_D1P	Primary	Undriven	Undriven	Driven	OFF
CNV_WT_D1N	Primary	Undriven	Undriven	Driven	OFF
CNV_WR_CLKP	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WR_CLKN	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WR_D0P	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WR_D0N	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WR_D1P	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WR_D1N	Primary	Undriven	Undriven	Powered (input)	OFF
CNV_WT_RCOMP	Primary	Undriven	Undriven	Driven	OFF

[1] Reset reference for primary well pins is PMC_RSMRST_N.

16.6 Functional Description

The main blocks of the integrated Connectivity solution are partitioned according to the following:

- **Connectivity Controller IP** contains:
  - Interfaces to the PCH
  - 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 PCH
  - Interfaces to the LTE modem via PCH 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.

17.0 PCI Express* (PCIe*)

This chapter provides information on the following topics:

- Features Supported
- Signal Description
- I/O Signal Planes and States
- PCI Express* Port / Controller Mapping

17.1 Features Supported

- Interrupt Generation
- PCI Express* Power Management
- Latency Tolerance Reporting (LTR)
- Dynamic Link Throttling
- Port 8xh Decode
- PCI Express* Separate Reference Clock with Independent Spread Spectrum Clocking (SRIS)
- Advanced Error Reporting
- Single Root I/O Virtualization (SR- IOV) Capability with Access Control Services (ACS) and Alternative Routing ID (ARI)
- SERR# Generation
- PCI Express* ExpressCard 1.0 module based hot-plug
- PCI Express* TX and RX Lane Polarity Inversion
- End-to-End PCI Express* Controller Lane Reversal
- Dynamic Link Width Negotiation as a Target
- Dynamic Speed Change
- 256B Maximum Data Payload Size
- PCIe* Subtractive Decode is not supported
  - PCI can still be supported via a PCIe*-to-PCI bridge. However, legacy PCI devices (such as PCMCIA or non-plug-and-play device) that need subtractive decode are not supported.
- Common RefClk RX Architecture support
- Precision Time Measurement (PTM)

17.1.1 Interrupt Generation

The root port generates interrupts on behalf of hot-plug, power management, link bandwidth management, Link Equalization Request and link error events, when enabled. These interrupts can either be pin-based, or can be Message Signal Interrupt (MSI), when enabled.

When an interrupt is generated using the legacy pin, the pin is internally routed to the SoC interrupt controllers. The pin that is driven is based upon the setting of the STRPFUSECFG.PXIP configuration registers.

Below table summarizes interrupt behavior for MSI and wire-modes. In the table "bits" refers to the hot-plug and PME interrupt bits.

Table 84. MSI Versus PCI IRQ Actions

Interrupt Register	Wire-Mode Action	MSI Action
All bits 0	Wire inactive	No action
One or more bits set to 1	Wire active	Send message
One or more bits set to 1, new bit gets set to 1	Wire active	Send message
One or more bits set to 1, software clears some (but not all) bits	Wire active	Send message
One or more bits set to 1, software clears all bits	Wire inactive	No action
Software clears one or more bits, and one or more bits are set on the same clock	Wire active	Send message

17.1.2 PCI Express* Power Management

S3/S4/S5 Support

Software initiates the transition to S3/S4/S5 by performing an I/O write to the Power Management Control register in the SoC. After the I/O write completion has been returned to the processor, the Power Management Controller will signal each root port to send a PME_Turn_Off message on the downstream link. The device attached to the link will eventually respond with a PME_TO_Ack followed by sending a PM_Enter_L23 DLLP (Data Link Layer Packet) request to enter L23. The Express ports and Power Management Controller take no action upon receiving a PME_TO_Ack. When all the Express port links are in state L23, the Power Management Controller will proceed with the entry into S3/S4/S5.

Prior to entering S3, software is required to put each device into $D3_{HOT}$. When a device is put into $D3_{HOT}$, it will initiate entry into a L1 link state by sending a PM_Enter_L1 DLLP. Under normal operating conditions when the root ports sends the PME_Turn_Off message, the link will be in state L1. However, when the root port is instructed to send the PME_Turn_Off message, it will send it whether or not the link was in L1. Endpoints attached to the PCH can make no assumptions about the state of the link prior to receiving a PME_Turn_Off message.

Device Initiated PM_PME Message

When the system has returned to a working state from a previous low power state, a device requesting service will send a PM_PME message continuously, until acknowledged by the root port. The root port will take different actions depending upon whether this is the first PM_PME that has been received, or whether a previous message has been received but not yet serviced by the operating system.

If this is the first message received (RSTS.PS), the root port will set RSTS.PS, and log the PME Requester ID into RSTS.RID. If an interrupt is enabled using RCTL.PIE, an interrupt will be generated. This interrupt can be either a pin or an MSI if MSI is enabled using MC.MSIE.

If this is a subsequent message received (RSTS.PS is already set), the root port will set RSTS.PP. No other action will be taken.

When the first PME event is cleared by software clearing RSTS.PS, the root port will set RSTS.PS, clear RSTS.PP, and move the requester ID into RSTS.RID.

If RCTL.PIE is set, an interrupt will be generated. If RCTL.PIE is not set, a message will be sent to the power management controller so that a GPE can be set. If messages have been logged (RSTS.PS is set), and RCTL.PIE is later written from a 0b to a 1b, an interrupt will be generated. This last condition handles the case where the message was received prior to the operating system re-enabling interrupts after resuming from a low power state.

SMI/SCI Generation

Interrupts for power management events are not supported on legacy operating systems. To support power management on non-PCI Express aware operating systems, PM events can be routed to generate SCI. To generate SCI, MPC.PMCE must be set. When set, a power management event will cause SMSCS.PMCS to be set.

Additionally, BIOS workarounds for power management can be supported by setting MPC.PMME. When this bit is set, power management events will set SMSCS.PMMS, and SMI# will be generated. This bit will be set regardless of whether interrupts or SCI is enabled. The SMI# may occur concurrently with an interrupt or SCI.

Latency Tolerance Reporting (LTR)

The root port supports the extended Latency Tolerance Reporting (LTR) capability. LTR provides a means for device endpoints to dynamically report their service latency requirements for memory access to the root port. Endpoint devices should transmit a new LTR message to the root port each time its latency tolerance changes (and initially during boot). The PCH uses the information to make better power management decisions. The processor uses the worst case tolerance value communicated by the PCH to optimize C-state transitions. This results in better platform power management without impacting endpoint functionality.

NOTE

Endpoint devices that support LTR must implement the reporting and enable mechanism detailed in the PCI-SIG “Latency Tolerance Reporting Engineering Change Notice” (www.pcisig.com).

17.1.3 Dynamic Link Throttling

Root Port supports dynamic link throttling as a mechanism to help lower the overall component power, ensuring that the component never operates beyond the thermal limit of the package. Dynamic link throttling is also used as a mechanism for ensuring that the ICC_{max} current rating of the voltage regulator is never exceeded. The target response time for this particular usage model is < 100 μs.

If dynamic link throttling is enabled, the link will be induced by the Root Port to enter TxL0s and RxL0s based on the throttle severity indication received. To induce the link into TxL0s, new TLP requests and opportunistic flow control update will be blocked. Eventually, in the absence of TLP and DLLP requests, the transmitter side of the link will enter TxL0s.

The periodic flow control update, as required by the PCI Express Base Specification is not blocked. However, the flow control credit values advertised to the component on the other side of the link will not be incremented, even if the periodic flow control update packet is sent. Once the other component runs out of credits, it will eventually enter TxL0s, resulting in the local receiver entering RxL0s.

Each of the Root Ports receives four throttle severity indications; T0, T1, T2, and T3. The throttling response for each of the four throttle severity levels can be independently configured in the Root Port TNPT.TSLxM register fields. This allows the duty cycle of the Throttling Window to be varied based on the severity levels, when dynamic link throttling is enabled.

A Throttling Window is defined as a period of time where the duty cycle of throttling can be specified. A Throttling Window is sub-divided into a Throttling Zone and a Non-Throttling Zone. The period of the Throttling Zone is configurable through the TNPT.TT field. Depending on the throttle severity levels, the throttling duration specified by the TNPT.TT field will be multiplied by the multipliers configurable through TNPT.TSLxM.

The period of the Throttling Window is configurable through the TNPT.TP field. The Throttling Window is always referenced from the time a new Throttle State change indication is received by the Root Port or from the time the throttling is enabled by the configuration register. The Throttling Window and Throttling Zone timers continue to behave the same as in L0 or L0s even if the link transitions to other LTSSM states, except for L1, L23_Rdy and link down. For L1 case, the timer is allowed to be stopped and hardware is allowed to re-start the Throttling Window and the corresponding Throttling Zone timers on exit from L1.

17.1.4 Port 8xh Decode

The PCIe* root ports will explicitly decode and claim I/O cycles within the 80h – 8Fh range when MPC.P8XDE is set. The claiming of these cycles are not subjected to standard PCI I/O Base/Limit and I/O Space Enable fields. This allows a POST-card to be connected to the Root Port either directly as a PCI Express device or through a PCI Express to PCI bridge as a PCI card.

Any I/O reads or writes will be forwarded to the link as it is. The device will need to be able to return the previously written value, on I/O read to these ranges. BIOS must ensure that at any one time, no more than one Root Port is enabled to claim Port 8xh cycles.

17.1.5 Separate Reference Clock with Independent SSC (SRIS)

The current PCI-SIG “PCI Express* External Cabling Specification” (www.pcisig.com) defines the reference clock as part of the signals delivered through the cable. Inclusion of the reference clock in the cable requires an expensive shielding solution to meet EMI requirements.

The need for an inexpensive PCIe* cabling solution for PCIe* SSDs requires a cabling form factor that supports non-common clock mode with spread spectrum enabled, such that the reference clock does not need to be part of the signals delivered through the cable. This clock mode requires the components on both sides of a link to tolerate a much higher ppm tolerance of ~5600 ppm compared to the PCIe* Base Specification defined as 600 ppm.

Soft straps are needed as a method to configure the port statically to operate in this mode. This mode is only enabled if the SSD connector is present on the motherboard, where the SSD connector does not include the reference clock. No change is being made to PCIe* add-in card form factors and solutions.

ASPM L0s is not supported in this form factor. The L1 exit latency advertised to software would be increased to 10 us. The root port does not support Lower SKP Ordered Set generation and reception feature defined in SRIS ECN.

17.1.6 Advanced Error Reporting

The PCI Express* Root Ports each provide basic error handling, as well as Advanced Error Reporting (AER) as described in the latest PCI Express Base Specification.

17.1.7 Single- Root I/O Virtualization (SR- IOV)

Alternative Routing ID Interpretation (ARI) and Access Control Services (ACS) are supported as part of the complementary technologies to enable SR-IOV capability.

Alternative Routing- ID Interpretation (ARI)

Alternative Routing-ID Interpretation (ARI) is a mechanism that can be used to extend the number of functions supported by a multi-function ARI device connected to the Root Port, beyond the conventional eight functions.

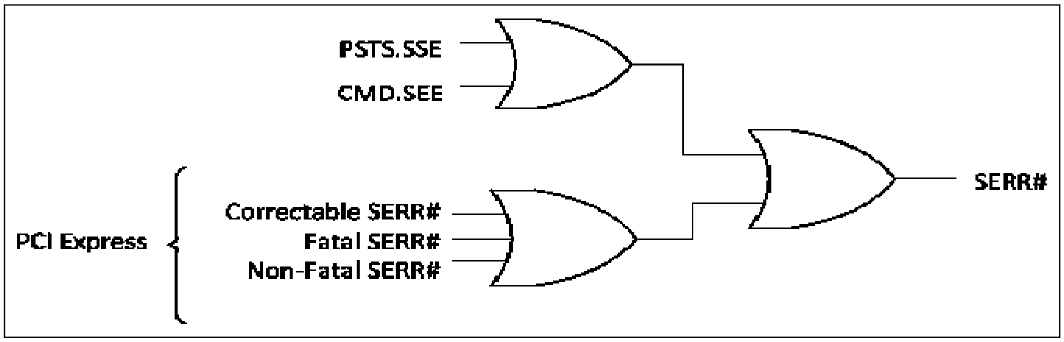
Access Control Services (ACS)

ACS is defined to control access between different Endpoints and between different Functions of a multi-function device. ACS defines a set of control points to determine whether a TLP should be routed normally, blocked, or redirected.

17.1.8 SERR# Generation

SERR# may be generated using two paths—through PCI mechanisms involving bits in the PCI header, or through PCI Express* mechanisms involving bits in the PCI Express capability structure.

Figure 36. Generation of SERR# to Platform



17.1.9 Hot-Plug

All PCIe* Root Ports support Express Card 1.0 based hot-plug that performs the following:

- Presence Detect and Link Active Changed Support
- Interrupt Generation Support

Presence Detection

When a module is plugged in and power is supplied, the physical layer will detect the presence of the device, and the root port sets SLSTS.PDS and SLSTS.PDC. If SLCTL.PDE and SLCTL.HPE are both set, the root port will also generate an interrupt.

When a module is removed (using the physical layer detection), the root port clears SLSTS.PDS and sets SLSTS.PDC. If SLCTL.PDE and SLCTL.HPE are both set, the root port will also generate an interrupt.

SMI/SCI Generation

Interrupts for power-management events are not supported on legacy operating systems. To support power-management on non-PCI Express aware operating systems, power management events can be routed to generate SCI. To generate SCI, MPC.HPCE must be set. When set, enabled hot-plug events will cause SMSCS.HPCS to be set.

Additionally, BIOS workarounds for hot-plug can be supported by setting MPC.HPME. When this bit is set, hot-plug events can cause SMI status bits in SMSCS to be set. Supported hot-plug events and their corresponding SMSCS bit are:

- Presence Detect Changed – SMSCS.HPPDM
- Link Active State Changed – SMSCS.HPLAS

When any of these bits are set, SMI# will be generated. These bits are set regardless of whether interrupts or SCI is enabled for hot-plug events. The SMI# may occur concurrently with an interrupt or SCI.

17.1.10 PCI Express* Lane Polarity Inversion

The PCI Express* Base Specification requires polarity inversion to be supported independently by all receivers across a Link—each differential pair within each Lane of a PCIe* Link handles its own polarity inversion. Polarity inversion is applied, as needed, during the initial training sequence of a Lane. In other words, a Lane will still function correctly even if a positive (Tx+) signal from a transmitter is connected to the negative (Rx-) signal of the receiver. Polarity inversion eliminates the need to untangle a trace route to reverse a signal polarity difference within a differential pair and no special configuration settings are necessary in the PCH to enable it. It is important to note that polarity inversion does not imply direction inversion or direction reversal; that is, the Tx differential pair from one device must still connect to the Rx differential pair on the receiving device, per the PCIe* Base Specification. Polarity Inversion is not the same as "PCI Express* Controller Lane Reversal".

17.1.11 PCI Express* Controller Lane Reversal

For each PCIe* Controller we support end-to-end lane reversal across the four lanes mapped to a controller for the two motherboard PCIe* configurations listed below. Lane Reversal means that the most significant lane of a PCIe* Controller is swapped with the least significant lane of the PCIe* Controller while the inner lanes get swapped to preserve the data exchange sequence (order).

NOTE

Lane Reversal Supported Motherboard PCIe* Configurations = 1x4, 2x1+1x2, and 2x2

- The 2x1+1x2 configuration is enabled by setting the PCIe* Controller soft straps to 1x2+2x1 with Lane Reversal Enabled

NOTE

PCI Express* Controller Lane Reversal is not the same as PCI Express* Lane Polarity Inversion

17.1.12 Precision Time Measurement (PTM)

Hardware protocol for precise coordination of events and timing information across multiple upstream and downstream devices using Transaction Layer Protocol (TLP) Message Requests. Minimizes timing translation errors resulting in the increased coordination of events across multiple components with very fine precision.

All of the PCH PCIe* Controllers and their assigned Root Ports support PTM where each Root Port can have PTM enabled or disabled individually from one another.

17.2 Signal Description

Name	Type	Description
PCIE[8:1]_TXP **PCIE**[8:1]**_TXN**	O	PCI Express* Differential Transmit Pairs These are PCI Express* based outbound high-speed differential signals
PCIE[8:1]_RXP **PCIE**[8:1]**_RXN**	I	PCI Express* Differential Receive Pairs These are PCI Express* based inbound high-speed differential signals
PCIE_RCOMPP **PCIE_RCOMPN**	I	Impedance Compensation Inputs

17.3 I/O Signal Planes and States

Table 85. Power Plane and States for PCI Express* Signals

Signal Name	Type	Power Plane	During Reset[1]	Immediately After Reset[1]	S3/S4/S5	Deep Sx
PCIE_TXP **PCIE_TXN**	O	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
PCIE_RXP **PCIE_RXN**	I	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
PCIE_RCOMPP **PCIE_RCOMPN**	I	Primary	Undriven	Undriven	Undriven	OFF

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

17.4 PCI Express* Port / Controller Mapping

Controller	Signal Name (External Port No.)	PCIe* Port No. Reported in Root Port Registers (Controller Port No.)	Default Root Port Device: Function
SPB (4x1, 2x2, 1x2 + 2x1, or 1 x4)	PCIE_1	5	Device 28:Function 4
	PCIE_2	6	Device 28:Function 5
	PCIE_3	7	Device 28:Function 6
	PCIE_4	8	Device 28:Function 7
SPA (4x1, 2x2 or 1x2 + 2x1)	PCIE_5	1	Device 28:Function 0
	PCIE_6	2	Device 28:Function 1
	PCIE_7	3	Device 28:Function 2
	PCIE_8	4	Device 28:Function 3

18.0 Universal Serial Bus (USB)

The PCH implements an xHCI USB controller which provides support for up to eight USB 2.0 signal pairs, four USB 3.0 and two USB 3.1 signal pairs.

The xHCI controller supports wake up from sleep states S1-S4. The eXtensible Host Controller (xHCI) supports up to 64 devices for a max number of 2048 Asynchronous endpoints (Control/Bulk) or max number of 128 Periodic Endpoints (Interrupt/ isochronous).

Each walk-up USB 3.2 capable port contains one USB 2.0 signal pair and one USB 3.2 signal pair.

The USB subsystem also supports Dual Role Capability. The xHCI is paired with a standalone eXtensible Device Controller Interface (xDCI) to provide dual role functionality. There is only one endpoint supported.

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. While the port is mapped to the device controller, the host controller Rx detection must always indicate a disconnected port.

18.1 Features Supported

- Device
  - D0i2 and D0i3 power gating
  - Wake capable on host initiated wakes when system is in S0i3, Sx
  - Available on all ports
- Port Routing Control for Dual Role Capability

Table 86. USB Bandwidth Information

USB Interface	
Category	**Description**
USB 3.2 Gen 2x1	2 (1x Dual Role Configurable on any one port)
Peak USB 3.2 Gen 2x1 Speed	10 Gb/s (Host Role), 5 Gb/s (Device Role)
USB 3.2 Gen 1x1 Port	4 (1x Dual Role Configurable on any one port)
Peak USB 3.2 Gen 1x1 Speed	5 Gb/s (Host Role), 2.5 Gb/s (Device Role)
USB 2.0 Port	8 (1x Dual Role Configurable on any one port)
Direct Connect Interface (DCI)	USB 3.x USB 2.0
Peak USB 2.0 Speed	480 Mb/s

18.2 USB Controllers Overview

Extensible Host Controller Interface (xHCI) is the interface specification that defines Host Controller for Universal Serial Bus (USB), which is capable of interfacing with USB 1.x, 2.x, and 3.x compatible devices.

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

Extensible Device Controller Interface (xDCI) is the interface specification that defines Device Controller for Universal Serial Bus (USB), which is capable of interfacing with USB 1.x, 2.x, and 3.x compatible devices

In case that the computer is connected as a device (example, tablet connected to desktop) to other computer then the xDCI controller will be activated inside the device will talk to the Host at the other computer.

Table 87. USB Specification

Protocol Name	Data Rate
USB 2.0 Low - Speed	1.5 Mbps
USB 2.0 Full - Speed	12 Mbps
USB 2.0 High - Speed	480 Mbps
USB 3.2 Gen 1x1	5 Gbps
USB 3.2 Gen2x1	10 Gbps (xHCI only)

18.3 Signal Description

Name	Type	Description
USB31_1_RXN **USB31_1_RXP**	I	**USB 3.2 Differential Receive Pair 1:** These are USB 3.2-Gen 2x1 differential signals for Port #1 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals.
USB31_1_TXP **USB31_1_TXN**	O	**USB 3.2 Differential Transmit Pair 1:** These are USB 3.2-Gen 2x1 differential signals for Port #1 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals.
USB31_2_RXN **USB31_2_RXP**	I	**USB 3.2 Differential Receive Pair 2:** These are USB 3.2-Gen 2x1 differential signals for Port #2 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals.
USB31_2_TXN **USB31_2_TXP**	O	**USB 3.2 Differential Transmit Pair 2:** These are USB 3.2-Gen 2x1 differential signals for Port #2 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals.
SATA_1_RXP**/** **USB30_4_RXP/** PCIE_8_RXP SATA_1_RXN**/** **USB30_4_RXN/** PCIE_8_RXN	I	**USB 3.2 Differential Receive Pair 4:** These are USB 3.2-Gen 1x1 differential signals for Port #4 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* or SATA.

continued...

Name	Type	Description
SATA_1_TXP**/ USB30_4_TXP/**PCIE_8_TXP SATA_1_TXN**/ USB30_4_TXN/** PCIE_8_TXN	O	**USB 3.2 Differential Transmit Pair 4:** These are USB 3.2-Gen 1x1 differential signals for Port #4 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* or SATA.
PCIE_6_RXP**/ USB30_1_RXP** PCIE_6_RXN**/ USB30_1_RXN**	I	**USB 3.2 Differential Receive Pair 1:** These are USB 3.2-Gen 1x1 differential signals for Port #5 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 6.
PCIE_6_TXP/USB30_1_TXP, PCIE_6_TXN**/ USB30_1_TXN**	O	**USB 3.2 Differential Transmit Pair 1:** These are USB 3.2-Gen 1x1 differential signals for Port #5 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 6.
PCIE_5_RXP**/ USB30_2_RXP,** PCIE_5_RXN**/ USB30_2_RXN**	I	**USB 3.2 Differential Receive Pair 2:** These are USB 3.2-Gen 1x1 differential signals for Port #6 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 5.
PCIE_5_TXP**/USB30_2_TXP ,**PCIE_5_TXN**/ USB30_2_TXN**	O	**USB 3.2 Differential Transmit Pair 2:** These are USB 3.2-Gen 1x1 differential signals for Port #6 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 5.
PCIE_4_RXP**/ USB30_3_RXP,** PCIE_5_RXN**/ USB30_3_RXN**	I	**USB 3.2 Differential Receive Pair 3:** These are USB 3.2-Gen 1x1 differential signals for Port #6 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 5.
PCIE_4_TXP**/USB30_3_TXP ,**PCIE_4_TXN**/ USB30_3_TXN**	O	**USB 3.2 Differential Transmit Pair 3:** These are USB 3.2-Gen 1x1 differential signals for Port #6 and the xHCI. It should be map to a USB connector with one of the OC (overcurrent) signals. *Note:* Use FITC to set the soft straps that select this port as PCIe* Port 5.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.

continued...

Name	Type	Description
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
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 and should map to a USB connector with one of the OC (overcurrent) signals.
GP_A18/**USB_OC0_N**	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GP_A12/**USB_OC1_N**	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GP_A13/**USB_OC2_N**	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GP_A14/**USB_OC3_N**	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB_VBUSSENSE	I	VBUS Sense for USB Device mode. Refer to OTG 3.0 specification for the sensing threshold voltage spec. • This HW signal is not used on the PCH for USB device mode functionality. This signal should be connected to ground.
USB_ID	I	ID detect for USB Device mode. • This HW signal is not used on the PCH for dual role mode selection. The switching of USB port role is done through the eSPI message from EC. This signal should be connected to ground.
USB2_RCOMP	I	USB Resistor Bias, analog connection points for an external resistor to ground.

18.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
USB2N_[8:1]	Internal Pull-down	14.25–24.8 kohm	1
USB2P_[8:1]	Internal Pull-down	14.25–24.8 kohm	1
USB_ID	Internal Weak Pull-up	14.25–24.8 kohm	If this signal is not in use, then the pin shall be connected directly to ground.
Note: 1. Series resistors (45 ohm ±10%)			

18.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S3/S4/S5	Deep Sx
USB31_[2:1]_RXN **USB31_[2:1]_RXP**	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB30_[4:1]_TXN **USB30_[4:1]_TXP**	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB2N_[8:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB2P_[8:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB_OC0_N	Primary	Undriven	Undriven	Undriven	OFF
USB_OC1_N	Primary	Undriven	Undriven	Undriven	OFF
USB_OC2_N	Primary	Undriven	Undriven	Undriven	OFF
USB_OC3_N	Primary	Undriven	Undriven	Undriven	OFF
USB_VBUSSENSE	Primary	Undriven	Undriven	Undriven	OFF
USB_ID[1]	Primary	Internal Pull-up	Undriven/ Internal Pull-up	Undriven/ Internal Pull-up	OFF
USB2_COMP	Primary	Undriven	Undriven	Undriven	OFF

Notes: 1. The USB_ID pin is pulled-up internally.
2. Reset reference for primary well pins is PMC_RSMRST_N and DSW well pins is DSW_PWROK.

19.0 Serial ATA (SATA)

The SATA controller support two modes of operation, AHCI mode using memory space. The SATA controller no longer supports IDE legacy mode using I/O space. Therefore, AHCI software is required. The PCH SATA controller supports the Serial ATA Specification, Revision 3.2.

19.1 Signals Description

Name	Type	Description
GP_E03/ SATA_0_DEVSLP	OD	**Serial ATA Port [0] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to Pull-up that's internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state.
GP_E07/ SATA_1_DEVSLP	OD	Serial ATA Port [1] Device Sleep: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to Pull-up that's internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state.
SATA_1_TXP/ USB30_4_TXP/ PCIE_8_TXP **SATA_1_TXN**/ USB30_4_TXN/ PCIE_8_TXN	O	Serial ATA Differential Transmit Pair 1: These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 8 and USB 3.2 Gen 1x1 signals.
SATA_1_RXP/ USB30_4_RXP/ PCIE_8_RXP **SATA_1_RXN**/ USB30_4_RXN/ PCIE_8_RXN	I	Serial ATA Differential Receive Pair 1: These inbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 8 and USB 3.2 Gen 1x1 signals.
SATA_0_TXP/ PCIE_7_TXP **SATA_0_TXN**/ PCIE_7_TXN	O	Serial ATA Differential Transmit Pair 0: These outbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 7.
SATA_0_RXP/ PCIE_7_RXP **SATA_0_RXN**/ PCIE_7_RXN	I	Serial ATA Differential Receive Pair 0: These inbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 7.
GP_E08/ SATA_0_GP	I	Serial ATA Port [0] General Purpose Inputs: When configured as SATAGP0, this is an input pin that is used as an interlock switch status indicator for SATA Port 0. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GP_E05/ SATA_LED_N	OD O	Serial ATA LED: This signal is an open-drain output pin driven during SATA command activity. It is to be connected to external circuitry that can provide the current to drive a platform LED. When active, the LED is on. When tri-stated, the LED is off.

19.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor type	Notes
SATA_[1:0]_RXN/P SATA_[1:0]_TXN/P	Internal pull-up	Internal Pull-Up Resistors are 15k-40k unless specified.

19.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[3]	Immediately After Reset[3]	S3/S4/S5	Deep Sx
SATA_[1:0]_TXP/N, SATA_[1:0]_RXP/N[3]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATA_LED_N	Primary	Undriven	Undriven	Undriven	OFF
SATA_[1:0]_DEVSLP	Primary	Undriven	Undriven	Driven Low	OFF
SATA_0_GP	Primary	Undriven	Undriven	Undriven	OFF

1. Pin defaults to GPIO mode. The pin state during and immediately after reset follows default GPIO mode pin state. The pin state for S0 to Deep Sx reflects assumption that GPIO Use Select register was programmed to native mode functionality. If GPIO Use Select register is programmed to GPIO mode, refer to Multiplexed GPIO (Defaults to GPIO Mode) section for the respective pin states in S0 to Deep Sx.
2. Reset reference for primary well pins is PMC_RSMRST_N.

19.4 Functional Description

- The SATA host controller (D23:F0) supports AHCI mode.
- The SATA controller does not support legacy IDE mode or combination mode.
- The SATA controller interacts with an attached mass storage device through a register interface that is compatible with an SATA AHCI host adapter. The host software follows existing standards and conventions when accessing the register interface and follows standard command protocol conventions.

19.4.1 SATA 6 Gb/s Support

The SATA controller is SATA 6 Gb/s capable and supports 6 Gb/s transfers with all capable SATA devices. The SATA controller also supports SATA 3 Gb/s and 1.5 Gb/s transfer capabilities.

19.4.2 SATA Feature Support

The SATA controller is capable of supporting all AHCI 1.3 and AHCI 1.3.1, refer Advanced Host Controller Interface Specification for current specification status available at: http://www.intel.com/content/www/us/en/io/serial-ata/ahci.html.

19.4.3 Power Management Operation

Power management of the PCH SATA controller and ports will cover operations of the host controller and the SATA link.

Power State Mappings

The D0 PCI Power Management (PM) state for device is supported by the PCH SATA controller.

SATA devices may also have multiple power states. SATA adopted 3 main power states from parallel ATA. The three device states are supported through ACPI. They are:

- **D0** – Device is working and instantly available.
- **D1** – Device enters when it receives a STANDBY IMMEDIATE command. Exit latency from this state is in seconds.
- **D3** – From the SATA device’s perspective, no different than a D1 state, in that it is entered using the STANDBY IMMEDIATE command. However, an ACPI method is also called which will reset the device and then cut its power.

Each of these device states are subsets of the host controller’s D0 state.

Finally, the SATA specification defines three PHY layer power states, which have no equivalent mappings to parallel ATA. They are:

- **PHY READY** – PHY logic and PLL are both on and in active state.
- **Partial** – PHY logic is powered up, and in a reduced power state. The link PM exit latency to active state maximum is 10 ns.
- **Slumber** – PHY logic is powered up, and in a reduced power state. The link PM exit latency to active state maximum is 10 ms.
- **Devslp** – PHY logic is powered down. The link PM exit latency from this state to active state maximum is 20 ms, unless otherwise specified by DETO in Identify Device Data Log page 08h (Refer SATA Rev3.2 Gold specification).

Since these states have much lower exit latency than the ACPI D1 and D3 states, the SATA controller specification defines these states as sub-states of the device D0 state.

Power State Transitions

Partial and Slumber State Entry/Exit

The partial and slumber states save interface power when the interface is idle. It would be most analogous to CLKRUN# (in power savings, not in mechanism), where the interface can have power saved while no commands are pending. The SATA controller defines PHY layer power management (as performed using primitives) as a driver operation from the host side, and a device proprietary mechanism on the device side. The SATA controller accepts device transition types, but does not issue any transitions as a host. All received requests from a SATA device will be ACKed.

When an operation is performed to the SATA controller such that it needs to use the SATA cable, the controller must check whether the link is in the Partial or Slumber states, and if so, must issue a COMWAKE to bring the link back online. Similarly, the SATA device must perform the same COMWAKE action.

NOTE

SATA devices shall not attempt to wake the link using COMWAKE/COMINIT when no commands are outstanding and the interface is in Slumber.

DEVSLP State Entry/Exit

Device Sleep (DEVSLP) is a host-controlled SATA interface power state. To support a hardware autonomous approach that is software agnostic Intel is recommending that BIOS configure the AHCI controller and the device to enable Device Sleep. This allows the AHCI controller and associated device to automatically enter and exit Device Sleep without the involvement of OS software.

To enter Device Sleep the link must first be in Slumber. By enabling HIPM (with Slumber) or DIPM on a Slumber capable device, the device/host link may enter the DevSleep Interface Power state.

The device must be DevSleep capable. Device Sleep is only entered when the link is in slumber, therefore when exiting the Device Sleep state, the device must resume with the COMWAKE out-of-band signal (and not the COMINIT out-of-band signal). Assuming Device Sleep was asserted when the link was in slumber, the device is expected to exit DEVSLP to the DR_Slumber state. Devices that do not support this feature will not be able to take advantage of the hardware automated entry to Device Sleep that is part of the AHCI 1.3.1 specification and supported by Intel platforms.

Device D1 and D3 States

These states are entered after some period of time when software has determined that no commands will be sent to this device for some time. The mechanism for putting a device in these states does not involve any work on the host controller, other then sending commands over the interface to the device. The command most likely to be used in ATA/ATAPI is the “STANDBY IMMEDIATE” command.

Host Controller D3$_{HOT}$ State

After the interface and device have been put into a low power state, the SATA host controller may be put into a low power state. This is performed using the PCI power management registers in configuration space. There are two very important aspects to note when using PCI power management.

1. When the power state is D3, only accesses to configuration space are allowed. Any attempt to access the memory or I/O spaces will result in master abort.
2. When the power state is D3, no interrupts may be generated, even if they are enabled. If an interrupt status bit is pending when the controller transitions to D0, an interrupt may be generated.

When the controller is put into D3, it is assumed that software has properly shut down the device and disabled the ports. Therefore, there is no need to sustain any values on the port wires. The interface will be treated as if no device is present on the cable, and power will be minimized.

When returning from a D3 state, an internal reset will not be performed.

Low Power Platform Consideration

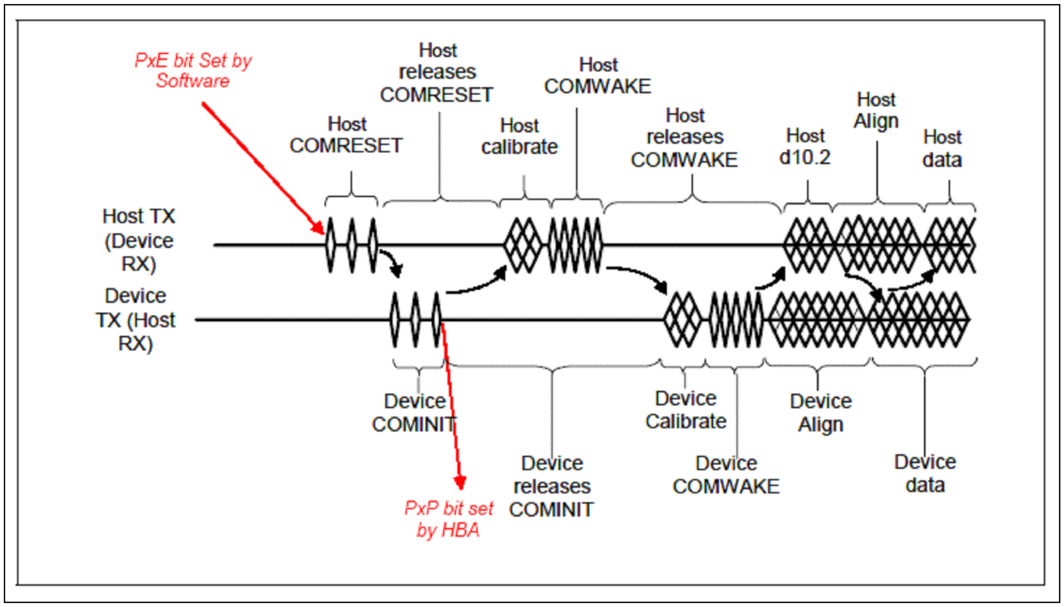
When low power feature is enabled, the Intel SATA controller may power off PLLs or OOB detection circuitry while in the Slumber link power state. As a result, a device initiated wake may not be recognized by the host. For example, when the low power feature is enabled it can prevent a Zero Power ODD (ZPODD) device from successfully communicating with the host on media insertion.

The SATA MPHY Dynamic Power Gating (PHYDPGEPx) can be enabled/disabled for each SATA ports.

19.4.4 SATA Device Presence

The flow used to indicate SATA device presence is shown in below figure. The 'PxE' bit refers to bits, depending on the port being checked and the 'PxP' bits refer to the bits, depending on the port being checked. If the PCS/PxP bit is set a device is present, if the bit is cleared a device is not present. If a port is disabled, software can check to view if a new device is connected by periodically re-enabling the port and observing if a device is present, if a device is not present it can disable the port and check again later. If a port remains enabled, software can periodically poll PCS.PxP to check if a new device is connected.

Figure 37. Port Enable/Device Present Bits Flow



19.4.5 SATA LED

The SATALED# output is driven whenever the BSY bit is set in any SATA port. The SATALED# is an active-low open-drain output. When SATALED# is low, the LED should be active. When SATALED# is high, the LED should be inactive.

19.4.6 Advanced Host Controller Interface (AHCI) Operation

The PCH SATA controller provides hardware support for Advanced Host Controller Interface (AHCI), a standardized programming interface for SATA host controllers developed through a joint industry effort. Platforms supporting AHCI may take advantage of performance features such as port independent DMA Engines—each device is treated as a master—and hardware-assisted native command queuing.

AHCI defines transactions between the SATA controller and software and enables advanced performance and usability with SATA. Platforms supporting AHCI may take advantage of performance features such as no master/slave designation for SATA devices—each device is treated as a master—and hardware assisted native command queuing. AHCI also provides usability enhancements such as hot-plug and advanced power management. AHCI requires appropriate software support (such as, an AHCI driver) and for some features, hardware support in the SATA device or additional platform hardware. Visit the Intel web site for current information on the AHCI specification.

The PCH SATA controller supports all of the mandatory features of the *Serial ATA Advanced Host Controller Interface Specification*, Revision 1.3.1 and many optional features, such as hardware assisted native command queuing, aggressive power management, LED indicator support, and hot-plug through the use of interlock switch support (additional platform hardware and software may be required depending upon the implementation).

NOTE

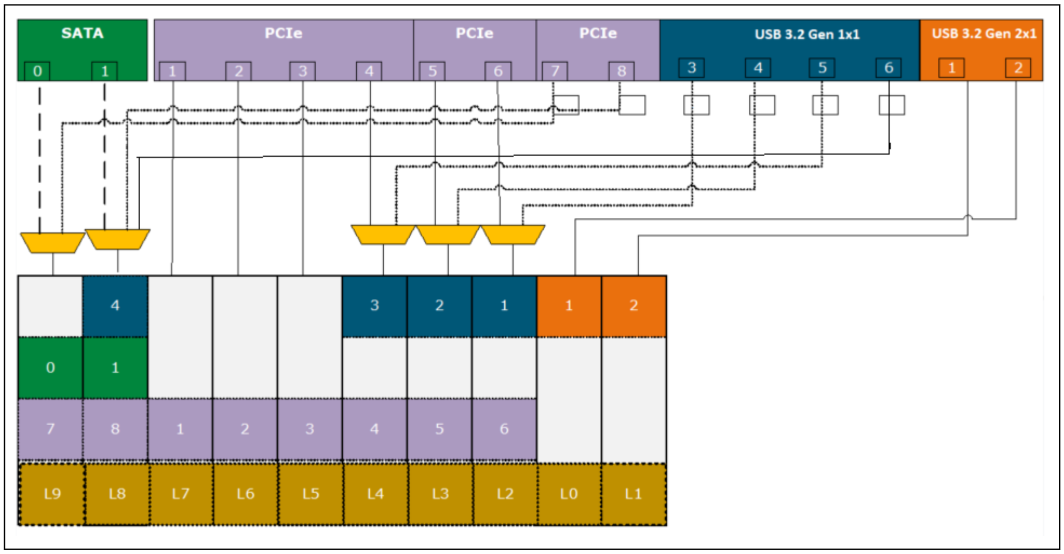
For reliable device removal notification while in AHCI operation without the use of interlock switches (surprise removal), interface power management should be disabled for the associated port. Visit the Intel web site for current information on the AHCI specification.

20.0 Flexible I/O

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

Figure below shows the High Speed I/O (HSIO) lane multiplexing.

Figure 38. HSIO Controller Port Configuration



The ten HSIO lanes on PCH supports the following configurations:

1. Up to 8PCIe* Lanes
2. Up to 2 SATA Lanes
3. Up to 6 USB 3.x Lanes

20.1 Flexible I/O Lane Selection

HSIO lane configuration and type is statically selected by soft straps, which are managed through the Flash Image Tool (FIT), available as part of Intel® CSE FW releases.

NOTE

It is the responsibility of the platform designers to configure the lane multiplexing and soft straps correctly without any conflict. The hardware behavior is undefined if this scenario ever happens.

21.0 Storage

This chapter provides information on the following topics:

- embedded Multi Media Card (eMMC*)
- I/O Signal Planes and States
- Secure Digital (SD)
- I/O Signal Planes and States

21.1 embedded Multi Media Card (eMMC*)

The eMMC* is a universal data storage and communication media. It is designed to cover a wide area of applications such as smart phones, tablets, computers, cameras, and so on. PCH supports only 1.8 V operating devices and PCH supports eMMC* version 5.1.

21.1.1 Features Supported

- HW Command Queuing support complaint to eMMC* v5.1 specification
- Support enhanced Strobe for HS400 mode @1.8 V
- Both ADMA2/DMA and Non-DMA mode of operation
- Transfers the data in 1 bit, 4 bit and 8 bit mode
- Support 64b address
- Cyclic Redundancy Check CRC7 for command and CRC16 for data integrity
- Support for Tx Path tuning and retention of DLL delay values

21.1.2 eMMC Signals Description

Name	Type	Description
EMMC_CMD	I/O	eMMC* Command/Response
EMMC_DATA[7:0]	I/O	eMMC* Data [7:0]
EMMC_RCLK	I	eMMC* Receive Clock
EMMC_CLK	O	eMMC* Clock
EMMC_RCOMP	I/O	eMMC* Compensation (200 Ohm +/- 1 % pull down to ground)
EMMC_RESET_N	O	eMMC* Device/Media Reset

21.2 I/O Signal Planes and States

Signal Name	Power Well	During Reset[1]	Immediately after Reset[1]	S0/S3/S4/S5	Deep Sx
EMMC_DATA[7:0]	Primary	Undriven	Undriven	Undriven	OFF
EMMC_RCLK	Primary	Undriven	Undriven	Undriven	OFF
EMMC_CLK	Primary	Undriven	Undriven	Undriven	OFF
EMMC_CMD	Primary	Undriven	Undriven	Undriven	OFF
EMMC_RCOMP	Primary	Undriven	Undriven	Undriven	OFF
EMMC_RESET_N	Primary	Undriven	Undriven	Undriven	OFF
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

21.2.1 Functional Description

The Controller handles eMMC* Protocol at transmission, packing data, adding cyclic redundancy check (CRC), start/end bit, and checking for transaction format correctness. Main supported features are listed below.

The eMMC* main use case is to connect an on board external storage device.

eMMC* 5.1 Command Queuing

Command Queuing (CQ) definition for eMMC* includes new commands for issuing tasks to the device, for ordering the execution of previously issued tasks and for additional task management function. The host controller with CQ can queue up to 32 commands to the device and the device selects and indicates one of the queued commands to host for service.

The host controller implements additional logic for handling a door-bell based DMA for the 32 descriptor / task list and manages the entire CQ flow which includes:

- Fetch and send the tasks/commands to device using existing logic
- Maintains context of each queued command
- Periodically read the device queue status and indicates completion of task to SW.
- Implements interrupt coalescing to reduce burden on software ISR.

eMMC* 5.1 Enhanced Strobe

Enhanced Strobe Mode for HS400 improves upon the HS400 mode interface speed increase that was first defined in eMMC* version 5.0, by facilitating faster synchronization between the host and the device.

Refer JEDEC eMMC* 5.1 specification for additional information.

eMMC* Working Modes

eMMC* Mode	Data Rate	Clock Frequency	Max. Data Throughput
Compatibility	Single	0 – 25 MHz	25 MB/s
High Speed SDR	Single	0 – 25 MHz	25 MB/s

continued...

eMMC* Mode	Data Rate	Clock Frequency	Max. Data Throughput
High Speed DDR	Dual	0 – 25 MHz	50 MB/s
HS200	Single	0 - 200 MHz	200 MB/s
HS400	Dual	0 - 200 MHz	400 MB/s

21.3 Secure Digital (SD)

The SD controller is to connect to an external detachable storage and/or I/O devices. It supports SD specification version 3.0.

21.3.1 SDXC Signal Description

Group	Signal Name	Description
Clock	SD_SDIO_CLK	SDXC Clock signal
Data	SD_SDIO_D[3:0]	SDXC Data signals
Command	SD_SDIO_CMD	SDXC Command signal
Control	SD_SDIO_CD_N	SD Card detect
	SD_SDIO_WP	SD card write protect
Power Enable	SD_SDIO_PWR_EN_N	SD card power enable
RCOMP	SD3_RCOMP	Compensation (200 Ohm +/- 1% pull down to ground)

21.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
SD_SDIO_CMD	Primary	Undriven	Undriven	Undriven	OFF
SD_SDIO_D[3:0]	Primary	Undriven	Undriven	Undriven	OFF
SD_SDIO_CD_N	Primary	Undriven	Undriven	Undriven	OFF
SD_SDIO_CLK	Primary	Undriven	Undriven	Undriven	OFF
SD_SDIO_WP	Primary	Undriven	Undriven	Undriven	OFF
SD_SDIO_PWR_EN_N	Primary	Undriven	Driven	Undriven	OFF
SD3_RCOMP	Primary	Undriven	Undriven	Undriven	OFF
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

21.4.1 Features Supported

- Support SD 3.0 @ 1.8 V Signaling (UHS-1@ SDR 104/50/25/12 and DDR50)
- Support SD 3.0 @ 3.3 V Signaling (Default Speed Mode/High Speed Mode)
- Support Cyclic Redundancy Check CRC7 for command and CRC16 for data integrity
- Support Card Detection (Insertion / Removal) (SD memory card only)

- Support D1-line wake from S0/D0i3 (To enable SDIO v3.00 on SD Removable card slot)

21.4.2 Functional Description

The SD controller handles SD Protocol at transmission, packing data, adding cyclic redundancy check (CRC), start/end bit, and checking for transaction format correctness. The main use case for SD is to connect to an external detachable storage and /or I/O device. Both 1.8 V and 3.3 V signaling is supported. Additional information can be obtained from the specifications stated above.The following chart maps the working modes of SD.

Table 88. SD Working Modes

SD Mode	Data Rate	Clock Frequency	Maximum Data Throughput
Default Speed/SDR12	Single	0 – 25 MHz	12.5 MB/s
High Speed/SDR25	Single	0 – 50 MHz	25 MB/s
SDR50	Single	0 – 100 MHz	50 MB/s
DDR50	Dual	0 – 50 MHz	50 MB/s
SDR104	Single	0 – 208 MHz	104 MB/s

22.0 Serial Peripheral Interface (SPI)

The SoC provides two Serial Peripheral Interfaces (SPI). The SPI0 interface consists of 3 Chip Select signals. It is allowing up to two flash memory devices (SPI0_CS0# and SPI0_CS1#) and one TPM device (SPI0_CS2#) to be connected to the PCH. The SPI0 interface support either 1.8V or 3.3V.

22.1 Functional Description

This section contains information about FSPI for Flash and FSPI Support for TPM.

22.1.1 FSPI for Flash

The Serial Peripheral Interface (FSPI) supports two SPI flash devices via two chip select signals (FSPI_CS0_N and FSPI_CS1_N). 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.

PCH drives the interface clock at either 20 MHz, 33 MHz,50 MHz or 60 MHz(1 Load) and will function with flash devices that support at least one of these frequencies.

A SPI flash device supporting SFDP (Serial Flash Discovery Parameter) is required for all PCH designs. A SPI flash device with a valid descriptor MUST be attached directly to the PCH.

The PCH 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)

Operational Modes

The SPI Controller has one operational mode:

Descriptor Mode: This mode is required to enable the following SoC features:

- Converged Security Engine.
- Secure Boot.
- PCI Express* root port configuration.
- Supports for two SPI components using two separate chip select pins.
- Hardware enforced security restricting master accesses to different regions.
- Soft Strap region providing the ability to use Flash NVM to remove the need for pull-up/pull-down resistors for strapping processor features.

- Support for the SPI Fast Read instruction and frequencies greater than 20 MHz.
- Support for Single Input, Dual Output Fast reads.
- Use of standardized Flash instruction set.

SPI Flash Regions

In Descriptor Mode, the Flash is divided into separate regions.

Table 89. SPI Flash Regions

Region	Content
0	Flash Descriptor
1	BIOS
2	Converged Security Engine
3	RSVD
4	Platform Data
5	RSVD

Only two masters can access the regions: Host processor running BIOS code and the Intel® CSE (Converged Security Engine).

The Flash Descriptor and CSE 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

SPI flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4-KB or larger block. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the CSE and BIOS regions. The CSE region contains firmware to support CSE capabilities.

Table 90. Region Size Versus Erase Granularity of Flash Components

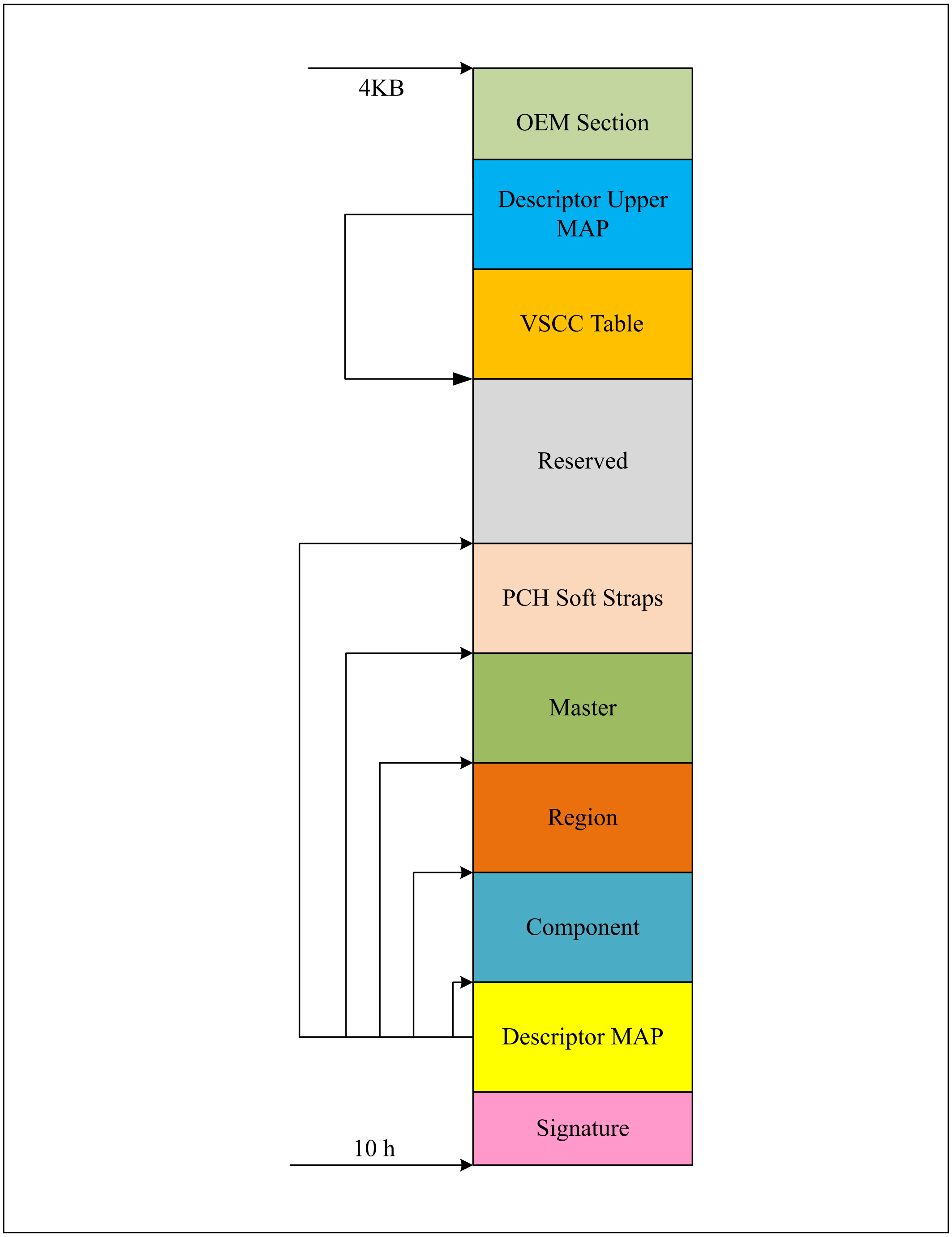
Region	Size with 4-KB Blocks	Size with 8-KB Blocks	Size with 64-KB Blocks
Descriptor	4 KB	8 KB	64 KB
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSE	Varies by Platform	Varies by Platform	Varies by Platform

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 SPI flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. The flash descriptor requires its own block at the bottom of memory (00h). 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 eleven sections as shown below.

Figure 39. Flash Descriptor Regions



- **OEM** Section is 256 bytes reserved at the top of the Flash Descriptor for use by OEM.
- **Descriptor Upper MAP** determines the length and base address of the Management Engine VSCC Table.

- **VSCC Table** holds the JEDEC ID and the VSCC information of the entire SPI Flash supported by the NVM image.
- **Reserved** region between the top of the processor strap section and the bottom of the OEM Section is reserved for future chipset usages.
- **PCH Soft Straps** section contains processor and PCH configurable parameters.
- **Master** region contains the security settings for the flash, granting read/write permissions for each region and identifying each master by a requestor ID.
- **Region** section points to the three other regions as well as the size of each region.
- **Component** section has information about the SPI flash in the system including: the number of components, density of each, invalid instructions (such as chip erase), and frequencies for read, fast read and write/erase instructions.
- **Descriptor Map** has pointers to the other five descriptor sections as well as the size of each.
- **Signature** selects Descriptor Mode as well as verifies if the flash is programmed and functioning. The data at the bottom of the flash (offset 10h) must be 0FF0A55Ah in order to be in Descriptor mode.

Descriptor Master Region

The master region defines read and write access setting for each region of the SPI device. The master region recognizes two masters: BIOS and CSE. Each master is only allowed to do direct reads of its primary regions.

Table 91. Region Access Control

Master Read/Write Access		
Region	**Processor and BIOS**	**Intel® CSE**
BIOS	Read/Write	N/A
CSE	N/A	Read/Write

Flash Descriptor CPU Complex Soft Strap Section

Region Name	Starting Address
Signature	10h
Component FCBA	30h
Regions FRBA	40h
Masters FMBA	80h
PCH Straps FPSBA	100h
MDTBA	C00h
PMC Straps	C14h
CPU Straps	C2Ch
Intel® CSE Straps	C3Ch
Register Init FIBA	340h

Flash Access

There are two types of accesses:

- Direct Access and
- Program Register Accesses.

Direct Access

- Masters are allowed to do direct read only of their primary region
- The BIOS or CSE virtual read address is converted into the SPI 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 Master Section
- Calculated Flash Linear Address must fall between primary region base/limit. If it does not, the cycle will not be run on the SPI bus, a completion with not data will be synthesized and returned with an Unsupported Request completion status and the AEL (Access Error Log) register error bit will be set
- Direct Write is not allowed with the exception of SPI TPM accesses
- Direct Read Cache contents are reset to 0's on a read from a different master

Program Register Access

- Program Register Accesses are not allowed to cross a 4-KB boundary and can not issue a command that might extend across two components
- Software programs the FLA corresponding to the region desired
  - Software must read devices Primary Region Base/Limit address to create FLA.

Register Access Security

- Only primary region masters can access the registers. If master ID is not valid, the cycle will not be run on the SPI bus, a a completion with no data will be synthesized and returned with an Unsupported Request completion status and the AEL (Access Error Log) register error bit will be set.

22.1.2 FSPI Support for TPM

The PCH's FSPI flash controller supports a discrete TPM on the platform via its dedicated FSPI_CS2_N signal. The platform must have no more than One TPM.

SPI controller supports accesses to SPI TPM at 20 MHz, 33 MHz and 60 MHz depending on the PCH soft strap. 20 MHz is the reset default, a valid PCH soft strap setting overrides the requirement for the 20 MHz. SPI TPM device must support a clock of 20 MHz. It may, but is not required to support a frequency greater than 20 MHz. The SPI controller does have an integrated interrupt signal for the TPM.

22.2 Signal Description

Name	Type	Description
FSPI_MOSI_IO0	Data	SPI serial output data from PCH to the SPI flash device. This Pin will also function as Input during Dual and Quad I/O operation
FSPI_MISO_IO1	Data	SPI serial input data from the SPI flash device to PCH. This Pin will also function as Output during Dual and Quad I/O operation
FSPI_IO2	Data	SPI serial Input/Output data to comprehend the support for the Quad I/O operation
FSPI_IO3	Data	SPI serial Input/Output data to comprehend the support for the Quad I/O operation
FSPI_CLK	Clock	SPI Clock output from PCH
FSPI_CS0_N	Chip Select	SPI chip select 0
FSPI_CS1_N	Chip Select	SPI chip select 1 signal is used as the second chip select, when 2 flash devices are used. Do not use when only one SPI flash is used.
FSPI_CS2_N	Chip Select	Chip Select 2 is dedicated to support TPM on SPI.

22.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value
FSPI_MOSI_IO0	Pull-Up	20k ± 30%
FSPI_MISO_IO1	Pull-Up	20k ± 30%
FSPI_IO2	Pull-Up	20k ± 30%
FSPI_IO3	Pull-Up	20k ± 30%
FSPI_CLK	Pull-up	20k ± 30%
FSPI_CS0_N	Pull-Up	20k ± 30%
FSPI_CS1_N	Pull-Up	20k ± 30%
FSPI_CS2_N	Pull-Up	20k ± 30%

NOTE

The internal pull-up is disabled when PMC_RSMRST_N is asserted (during reset) and only enabled after PMC_RSMRST_N de-assertion

22.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset	S3/S4/S5
FSPI_MOSI_IO0	Primary	Hi-Z	Internal PU, then Driven Low	Driven Low
FSPI_MISO_IO1	Primary	Hi-Z	Internal Pull-up	Internal Pull-up

continued...

Signal Name	Power Plane	During Reset[1]	Immediately after Reset	S3/S4/S5
FSPI_IO2	Primary	Hi-Z	Internal Pull-up	Internal Pull-up
FSPI_IO3	Primary	Hi-Z	Internal Pull-up	Internal Pull-up
FSPI_CLK	Primary	Internal Pulldown	Driven Low	Driven Low
FSPI_CS0_N	Primary	Internal Pulldown	Driven High	Driven High
FSPI_CS1_N	Primary	Internal Pulldown	Driven High	Driven High
FSPI_CS2_N	Primary	Internal Pulldown	Driven High	Driven High
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.				

23.0 Intel® Serial I/O Generic SPI (GSPI) Controllers

The PCH implements three generic SPI interfaces to support devices that uses serial protocol for transferring data.

Each interface consists of a clock (CLK), two chip selects (CS) and 2 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 master mode only

NOTE

Slave mode is not supported.

Table 92. Acronyms

Acronyms	Description
GSPI	Generic Serial Peripheral Interface
LTR	Latency Tolerance Reporting

23.1 Signal Description

Name	Type	Description
GSPI0_CS0_N	O	Generic SPI 0 Chip Select 0
GSPI0_CS1_N	O	Generic SPI 0 Chip Select 1
GSPI0_CLK	O	Generic SPI 0 Clock
GSPI0_MISO	I	Generic SPI 0 MISO
GSPI0_MOSI	O	Generic SPI 0 MOSI • This signal is also utilized as a strap. Refer the pin strap section for more info.
GSPI1_CS0_N	O	Generic SPI 1 Chip Select 0
GSPI1_CS1_N	O	Generic SPI 1 Chip Select 1
GSPI1_CLK	O	Generic SPI 1 Clock
GSPI1_MISO	I	Generic SPI 1 MISO

continued...

Name	Type	Description
GSPI1_MOSI	O	Generic SPI 1 MOSI • This signal is also utilized as a strap. Refer the pin strap section for more info.
GSPI2_CS0_N	O	Generic SPI 2 Chip Select 0
GSPI2_CS1_N	O	Generic SPI 2 Chip Select 1
GSPI2_CLK	O	Generic SPI 2 Clock
GSPI2_MISO	I	Generic SPI 2 MISO
GSPI2_MOSI	O	Generic SPI 2 MOSI

23.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
GSPI0_MOSI	Pull Down	20k ± 30%	The integrated pull down is disabled after PCH_PWROK assertion
GSPI1_MOSI	Pull Down	20k ± 30%	The integrated pull down is disabled after PCH_PWROK assertion
GSPI2_MOSI	Pull Down	20k ± 30%	The integrated pull down is disabled after PCH_PWROK assertion
GSPI0_MISO	Pull Down	20k ± 30%	
GSPI1_MISO	Pull Down	20k ± 30%	
GSPI2_MISO	Pull Down	20k ± 30%	

23.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
GSPI0_CS0_N, GSPI0_CS1_N, GSPI1_CS0_N, GSPI1_CS1_N, GSPI2_CS0_N, GSPI2_CS1_N	Primary	Undriven	Undriven	Undriven	OFF
GSPI2_CLK, GSPI1_CLK, GSPI0_CLK	Primary	Undriven	Undriven	Undriven	OFF
GSPI2_MISO, GSPI1_MISO, GSPI0_MISO	Primary	Undriven	Undriven	Undriven	OFF
GSPI2_MOSI, GSPI1_MOSI, GSPI0_MOSI	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF

Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.

23.4 Functional Description

This section provides information on the following topics:

- Controller Overview
- DMA Controller
- Reset
- Power Management
- Interrupts
- Error Handling

23.4.1 Controller Overview

The generic SPI controllers can only be set to operate as a master.

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 PCH 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 master, 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 PCH 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.

23.4.2 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.

23.4.3 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.

23.4.4 Power Management

Device Power Down Support

In order to power down peripherals connected to the PCH 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 PCH 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.

23.4.5 Interrupts

GSPI 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 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.

All interrupts are active high and their behavior is level interrupt.

23.4.6 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.

24.0 Enhanced Serial Peripheral Interface (eSPI)

The PCH provides the Enhanced Serial Peripheral Interface (eSPI) to support connection of an EC to the platform.

eSPI operates at 1.8 V only. This interface is not shared and distinct from the SPI interface used for flash device and TPM. The eSPI interface supports 20 MHz, 24 MHz, 30 MHz, 48 MHz, and 60 MHz and up to Quad Mode with one chip select.

24.1 Signal Description

Table 93. eSPI Signals

Signal Name	Group	Description
ESPI_IO[3:0]	Data	Bi-directional data signals used to transfer data between PCH and eSPI slave device
ESPI_CLK	Clock	eSPI Clock output from PCH
ESPI_CS_N	Control	eSPI chip select
ESPI_RESET_N	Control	eSPI reset signal
ESPI_ALERT[3:0]_N	Control	eSPI alert signal

24.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value
ESPI_CLK	Pull-down	20K +/- 30%
ESPI_IO[3:0]	Pull-up	20K +/- 30%
ESPI_ CS _N	Pull-up	20K +/- 30%

24.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
ESPI_CLK	Primary	Internal Pull-down	Driven Low	Driven Low	Off
ESPI_IO [3:0]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off
ESPI_ CS _N	Primary	Internal Pull-up	Driven High	Driven High	Off
ESPI_RESET_N	Primary	Driven Low	Driven High	Driven High	Off

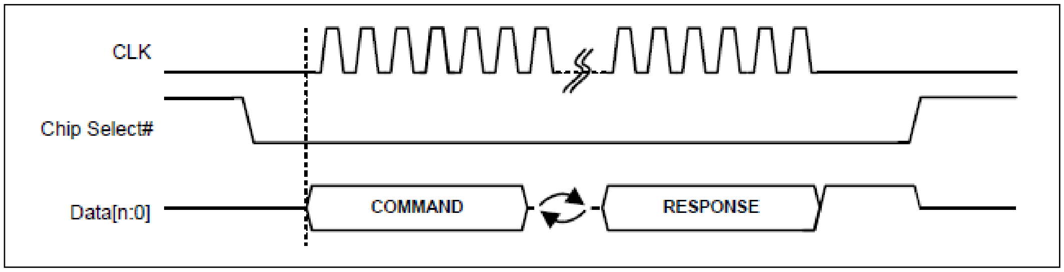
24.3.1 Operating Frequency

The eSPI controller supports 20 MHz, 24 MHz, 30 MHz, 48 MHz, and 60 MHz. A slave device can support frequencies lower than the recommended maximum frequency (60 MHz). In addition, the slave device must support a minimum frequency of 20 MHz for default (reset) communication between the Master and Slave device.

24.3.2 Protocols

Below is an overview of the basic eSPI protocol.

Figure 40. Basic eSPI Protocol



An eSPI transaction consists of a Command phase driven by the master, a turn-around phase (TAR), and a Response phase driven by the slave.

A transaction is initiated by the PCH through the assertion of CS#, starting the clock and driving the command onto the data bus. The clock remains toggling until the complete response phase has been received from the slave.

The serial clock must be low at the assertion edge of the CS# while ESPI_RESET# has been de-asserted. The first data is driven out from the PCH while the serial clock is still low and sampled on the rising edge of the clock by the slave. Subsequent data is driven on the falling edge of the clock from the PCH and sampled on the rising edge of the clock by the slave. Data from the slave is driven out on the falling edge of the clock and is sampled on a falling edge of the clock by the PCH.

All transactions on eSPI are in multiple of 8 bits (one byte).

24.3.3 WAIT States from eSPI Slave

There are situations when the slave cannot predict the length of the command packet from the master (PCH). For non-posted transactions, the slave is allowed to respond with a limited number of WAIT states.

A WAIT state is a 1-byte response code. They must be the first set of response byte from the slave after the TAR cycles.

24.3.4 In-Band Link Reset

In case the eSPI link may end up in an undefined state (for example when a CRC error is received from the slave in a response to a Set_Configuration command), the PCH issues an In-Band Reset command that resets the eSPI link to the default configuration. This allows the controller to re-initialize the link and reconfigure the slave.

24.3.5 Slave Discovery

The controller does not perform discovery to confirm the presence of the slave connection.

24.3.6 PECI Over eSPI

When PECI Over eSPI is enabled, the eSPI device (i.e. EC) can access the processor PECI interface via eSPI controller.

The PECI bus may be connected to the PCH via the eSPI interface. The operation over eSPI is selected via a soft strap.

PECI over eSPI is not supported in Sx state. The connected eSPI device is not allowed to send the PECI command to eSPI in Sx states. More specifically, the device can only send PECI requests after Virtual Wire PLT_RST# de-assertion.

In S0ix, upon receiving a PECI command, the PMC will wake up the CPU from Cx and respond back once the data is available from CP

24.3.7 Multiple OOB Master

PCHs typically have multiple embedded processors such as the PMC and CSE. From an eSPI perspective, these are all classified as Out-of-Band (OOB) processors (as distinct from the Host processor). Since any such OOB processors may need to communicate with the eSPI device on the platform (e.g., EC. BMC), the eSPI controller implements dedicated OOB channel for each OOB processors including PMC and CSE to improve the interface performance and potentially enable new usage models.

24.3.8 Channels and Supported Transactions

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 slave is discovered through the GET_CONFIGURATION command issued by the PCH to the eSPI slave during initialization.

Below a table summarizes the eSPI channels and supported transactions.

Table 94. 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 D31:F0:** The eSPI controller duplicates the legacy 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 Slave (EC/SIO) Debug Device Accesses:** 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 Slave 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 PCH and eSPI slave. These signals are tunneled between the 2 components over eSPI.
- Serial IRQ Interrupts: Interrupts are tunneled from the eSPI slave to the PCH. Both edge and triggered interrupts are supported.

eSPI Virtual Wires (VW)

Below table summarizes the PCH virtual wires in eSPI mode.

Table 95. eSPI Virtual Wires (VW)

Virtual Wire	PCH Pin Direction	Reset Control	Pin Retained in PCH (For Use by Other Components)
SUS_STAT#	Output	ESPI_RST0_N	No
SUSPWRDNACK	Output	ESPI_RST0_N	No
PLTRST#	Output	ESPI_RST0_N	Yes
PME# (eSPI Peripheral PME)	Input	ESPI_RST0_N	N/A
WAKE#	Input	ESPI_RST0_N	No
SMI#	Input	PMC_PLTRST_N	N/A
SCI#	Input	PMC_PLTRST_N	N/A
RCIN#	Input	PMC_PLTRST_N	No
SLAVE_BOOT_LOAD_DONE	Input	ESPI_RST0_N	N/A
SLAVE_BOOT_LOAD_STATUS	Input	ESPI_RST0_N	N/A
HOST_RST_WARN	Output	PMC_PLTRST_N	N/A
HOST_RST_ACK	Input	PMC_PLTRST_N	N/A
OOB_RST_WARN	Output	ESPI_RST0_N	N/A
OOB_RST_ACK	Input	ESPI_RST0_N	N/A

continued...

Virtual Wire	PCH Pin Direction	Reset Control	Pin Retained in PCH (For Use by Other Components)
HOST_C10	Output	PMC_PLTRST_N	N/A
ERROR_NONFATAL	Input	ESPI_RST0_N	N/A
ERROR_FATAL	Input	ESPI_RST0_N	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 PCH eSPI controller will issue a message to the PCH interrupt controller when it receives an IRQ group in its VW packet, indicating a state change for that IRQ line number.

The eSPI slave 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 slave. 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:

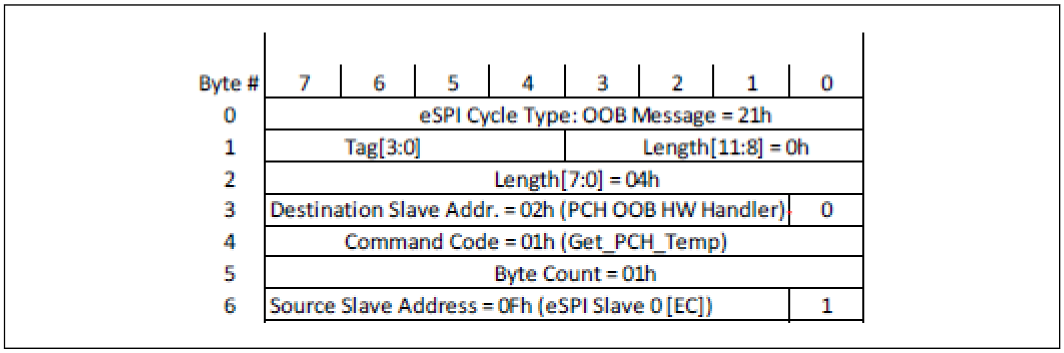
- **Tunnel PCH Temperature Data to the eSPI Slave:** The eSPI controller stores the PCH temperature data internally and sends it to the slave using a posted OOB message when a request is made to a specific destination address.
- **Tunnel PCH RTC Time and Date Bytes to the eSPI Slave:** the eSPI controller captures this data internally at periodic intervals from the PCH RTC controller and sends it to the slave device using a posted OOB message when a request is made to a specific destination address.

PCH Temperature Data Over eSPI OOB Channel

eSPI controller supports the transmitting of PCH thermal data to the eSPI slave. The thermal data consists of 1 byte of PCH temperature data that is transmitted periodically (~1 ms) from the thermal sensor unit.

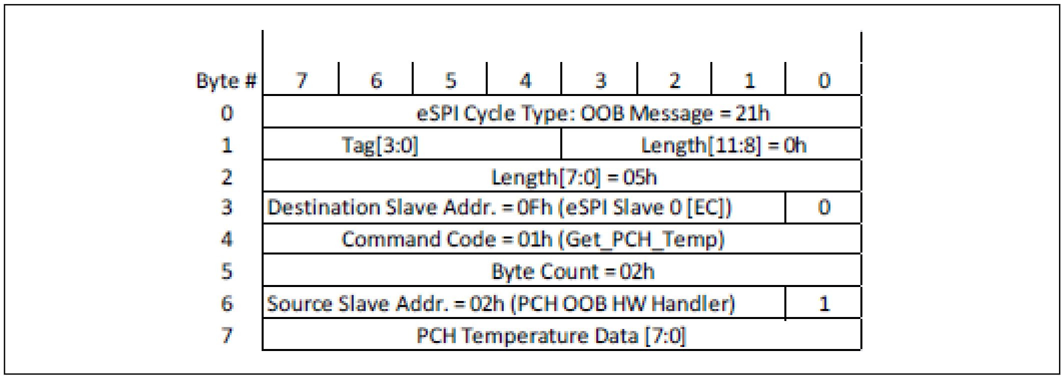
The packet formats for the temperature request from the eSPI slave and the PCH response back are shown in following two figures.

Figure 41. eSPI Slave Request to PCH for PCH 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] = 04h							
3	Destination Slave Addr. = 02h (PCH OOB HW Handler)							0
4	Command Code = 01h (Get_PCH_Temp)							
5	Byte Count = 01h							
6	Source Slave Address = 0Fh (eSPI Slave 0 [EC])							1

Figure 42. PCH Response to eSPI Slave with PCH 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 Slave Addr. = 0Fh (eSPI Slave 0 [EC])							0
4	Command Code = 01h (Get_PCH_Temp)							
5	Byte Count = 02h							
6	Source Slave Addr. = 02h (PCH OOB HW Handler)							1
7	PCH Temperature Data [7:0]							

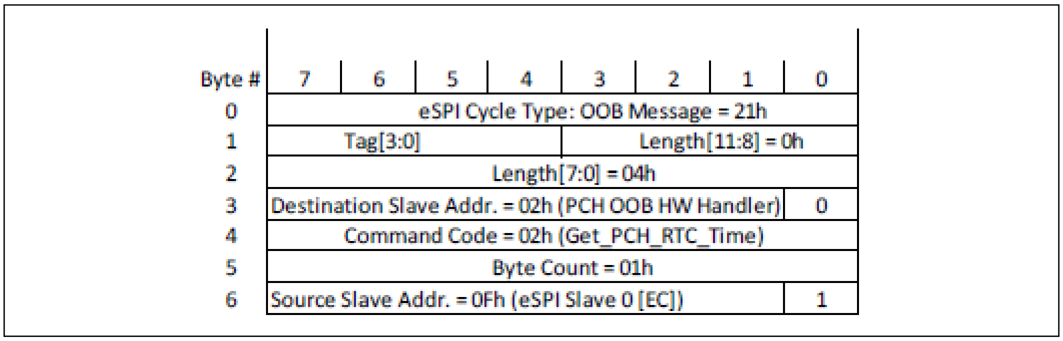
PCH RTC Time/Date to EC Over eSPI OOB Channel

The PCH eSPI controller supports the transmitting of PCH RTC time/date to the eSPI slave. This allows the eSPI slave to synchronize with the PCH 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.

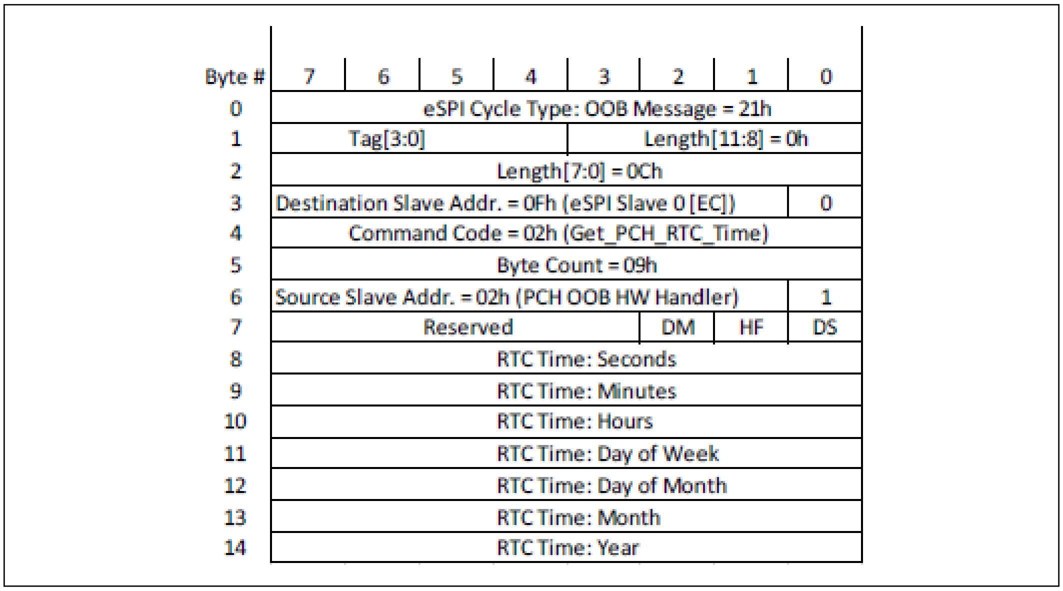
The packet formats for the RTC time/date request from the eSPI slave and the PCH response back to the device are shown in following two figures.

Figure 43. eSPI Slave Request to PCH for PCH 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	Destination Slave Addr. = 02h (PCH OOB HW Handler)							0
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 01h							
6	Source Slave Addr. = 0Fh (eSPI Slave 0 [EC])							1

Figure 44. PCH Response to eSPI Slave 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	Destination Slave Addr. = 0Fh (eSPI Slave 0 [EC])							0
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 09h							
6	Source Slave Addr. = 02h (PCH 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							

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.

25.0 Real Time Clock (RTC)

The PCH 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—keeping track of the time of day and storing 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 3 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 96. 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

25.1 Signal Description

Name	Type	Description
RTC_X1	I	**Crystal Input 1:** This signal is connected to the 32.768 kHz crystal (max 50K ohm ESR). If no external crystal is used, then RTCX1 can be driven with the desired clock rate. Maximum voltage allowed on this pin is 1.5V.
RTC_X2	O	**Crystal Input 2:** This signal is connected to the 32.768 kHz crystal (max 50K ohm ESR). If no external crystal is used, then RTCX2 must be left floating.
RTCRST_N	I	**RTC Reset:** When asserted, this signal resets register bits in the RTC well. *Notes:* 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. 2. In the case where the RTC battery is dead or missing on the platform, the RTCRST# pin must rise before the DSW_PWROK pin.
SRTCRST_N	I	**Secondary RTC Reset:** This signal resets the manageability register bits in the RTC well when the RTC battery is removed.

continued...

Name	Type	Description
		Notes: 1. The SRTCRST# input must always be high when all other RTC power planes are on. 2. In the case where the RTC battery is dead or missing on the platform, the SRTCRST# pin must rise before the DSW_PWROK pin. 3. SRTCRST# and RTCRST# should not be shorted together.

25.2 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
RTCRST_N	RTC	Undriven	Undriven	Undriven	Undriven
SRTCRST_N	RTC	Undriven	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

26.0 8254 Timers

The PCH contains two 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 1.193MHz periodic timer tick is generated off the 38.4/24 MHz 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.

Counter 2, Speaker Tone

This counter provides the speaker tone and is typically programmed for Mode 3 operation. The counter provides a speaker frequency equal to the counter clock frequency (1.193 MHz) divided by the initial count value. The speaker must be enabled by a write to port 061h (Refer NMI Status and Control ports).

26.1 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 two-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 all three counters. 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.

This table lists the six operating modes for the interval counters.

Table 97. 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.

26.2 Reading from the 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 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 two byte counts, two bytes must be read. The two 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.

26.2.1 Simple Read

The first method is to perform a simple read operation. The counter is selected through Port 40h (Counter 0) or 42h (Counter 2).

NOTE

Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of Counter 2, the count can be stopped by writing to the GATE bit in Port 61h.

26.2.2 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 two-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, some time 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.

26.2.3 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 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 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.

27.0 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 PCH 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.

27.1 References

Specification	Location
IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a	http://www.intel.com/content/dam/www/public/us/en/documents/technical-specifications/software-developers-hpet-spec-1-0a.pdf

27.2 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-microsecond 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 PCH’s 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 PMC_RSMRST_N.

27.3 Timer Off-load

The PCH supports a timer off-load feature that allows the HPET timers to remain operational during very low power S0 operational modes when the PCH’s 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 PCH's XTAL clock during boot time to an accuracy that ensures the error introduced by this off-load is less than 10 ppb (.000001%).

When the PCH's XTAL clock is active, the 64-bit counter will increment by one each cycle of the PCH's XTAL clock when enabled. When the PCH's XTAL clock is disabled, the timer is maintained using the RTC clock. The long-term (> 1 msec) frequency drift allowed by the HPET specification is 500 ppm. The off-load mechanism ensures that it contributes < 1ppm 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 in the PCH runs typically on the PCH's 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 PCH's XTAL MHz clock to shut down when it has no active comparators.

Theory of Operation

The Off-loadable Timer Block consists of a 64b fast clock counter and an 82b 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 microseconds 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 64b of the 82b counter, with the 18 LSBs set to zero. The actual transition through 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 64b of an 82b 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 64b 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.

27.4 Interrupt Mapping

The interrupts associated with the various timers have several interrupt mapping options. When reprogramming the HPET interrupt routing scheme (LEG_RT_CNF bit in the General Config Register), a spurious interrupt may occur. This is because the other source of the interrupt (8254 timer) may be asserted. Software should mask interrupts prior to clearing the LEG_RT_CNF bit.

Mapping Option #1 (Legacy Replacement Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is set. This forces the mapping found in the following table.

Table 98. Legacy Replacement Routing

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	In this case, the 8254 timer will not cause any interrupts
1	IRQ8	IRQ8	In this case, the RTC will not cause any interrupts.
2 and 3	Per IRQ Routing Field.	Per IRQ Routing Field	
4, 5, 6, 7	Not available	Not available	
Note: The Legacy Option does not preclude delivery of IRQ0/IRQ8 using processor interrupts messages.			

Mapping Option #2 (Standard Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is 0. Each timer has its own routing control. The interrupts can be routed to various interrupts in the 8259 or I/O APIC. A capabilities field indicates which interrupts are valid options for routing. If a timer is set for edge-triggered mode, the timers should not be shared with any legacy interrupts.

For the PCH, the only supported interrupt values are as follows:

Timer 0 and 1: IRQ20, 21, 22, and 23 (I/O APIC only).

Timer 2: IRQ11 (8259 or I/O APIC) and IRQ20, 21, 22, and 23 (I/O APIC only).

Timer 3: IRQ12 (8259 or I/O APIC) and IRQ 20, 21, 22, and 23 (I/O APIC only).

NOTE

Interrupts from Timer 4, 5, 6, 7 can only be delivered via direct FSB interrupt messages.

NOTE

System architecture changes since the HPET specification 1.0 was released have made some of the terminology used obsolete. In particular the reference to a Front Side Bus (FSB) has no relevance to current platforms, as this interface is no longer in use. For consistency with the HPET specification though, the FSB and specifically the FSB Interrupt Delivery terminology has been maintained. Where the specification refers to FSB, this should be read as ‘processor message interface’; independent of the physical attach mechanism.

Mapping Option #3 (Processor Message Option)

In this case, the interrupts are mapped directly to processor messages without going to the 8259 or I/O (x) APIC. To use this mode, the interrupt must be configured to edge-triggered mode. The Tn_PROCMSG_EN_CNF bit must be set to enable this mode.

When the interrupt is delivered to the processor, the message is delivered to the address indicated in the Tn_PROCMSG_INT_ADDR field. The data value for the write cycle is specified in the Tn_PROCMSG_INT_VAL field.

NOTE

The FSB interrupt deliver option has HIGHER priority and is mutually exclusive to the standard interrupt delivery option. Thus, if the TIMERn_FSB_EN_CNF bit is set, the interrupts will be delivered via the FSB, rather than via the APIC or 8259.

The FSB interrupt delivery can be used even when the legacy mapping is used.

For the Intel PCH HPET implementation, the direct FSB interrupt delivery mode is supported, besides via 8259 or I/O APIC.

27.5 Periodic Versus Non-Periodic Modes

Non-Periodic Mode

This mode can be thought of as creating a one-shot.

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 microseconds.

All of the timers support non-periodic mode.

Refer to Section 2.3.9.2.1 of the 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:

- Software clears the ENABLE_CNF bit to prevent any interrupts
- Software Clears the main counter by writing a value of 00h to it.
- Software sets the TIMER0_VAL_SET_CNF bit.
- 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. he reason for this is that supporting the periodic mode adds a significant amount of gates.

For the Intel PCH, only timer 0 will support the periodic mode. This saves a substantial number of gates.

27.6 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.

27.7 Interrupt Levels

Interrupts directed to the internal 8259s are active high. 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.

27.8 Handling Interrupts

Section 2.4.6 of the IA-PC HPET Specification describes handling interrupts.

27.9 Issues Related to 64-Bit Timers with 32-Bit Processors

Section 2.4.7 of the IA-PC HPET Specification describes issues related to 64-bit timers with 32-bit processors.

28.0 Intel® LPSS Inter-Integrated Circuit (I2C) Controllers

The PCH 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 I2C host controllers and do not incorporate a DMA controller. Therefore, I2C4 and I2C5 are restricted to operate in PIO mode only.

The I2C 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).
- 1.8 V or 3.3 V support (depending on the voltage supplied to the I2C signal group)
- Master 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 (t_{HD}; 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 master mode only. I^2C slave mode is not supported.
2. I^2C multi masters 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 99. Acronyms

Acronyms	Description
I2C	Inter-Integrated Circuit
PIO	Programmed Input/Output
SCL	Serial Clock Line
SDA	Serial Data Line

Table 100. References

Specification	Location
The I2C Bus Specification, Version 5	www.nxp.com/documents/user_manual/UM10204.pdf

28.1 Signal Description

Name	Type	Description
I2C0_SDA	I/OD	I^2C Link 0 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C0_SCL	I/OD	I^2C Link 0 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C1_SDA	I/OD	I^2C Link 1 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C1_SCL	I/OD	I^2C Link 1 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C2_SDA	I/OD	I^2C Link 2 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C2_SCL	I/OD	I^2C Link 2 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C3_SDA	I/OD	I^2C Link 3 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C3_SCL	I/OD	I^2C Link 3 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C4_SDA	I/OD	I^2C Link 4 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C4_SCL	I/OD	I^2C Link 4 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C4B_SDA	I/OD	I^2C Link 4 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C4B_SCL	I/OD	I^2C Link 4 Serial Clock Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C5_SDA	I/OD	I^2C Link 5 Serial Data Line External Pull-up resistor may be required depending on Bus Capacitance.
I2C5_SCL	I/OD	I^2C Link 5 Serial Clock Line

continued...

Name	Type	Description
		External Pull-up resistor may be required depending on Bus Capacitance.
Note: I2C4 and I2C4B are from the same I2C controller which are muxed on different set of pins, and as such only I2C4 **or** I2C4B can be used at any one time on a given platform.		

28.2 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
I2C[5:0]_SDA	Primary	Undriven	Undriven	Undriven	OFF
I2C[5:0]_SCL	Primary	Undriven	Undriven	Undriven	OFF
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

28.3 Functional Description

This section provides information on the following topics:

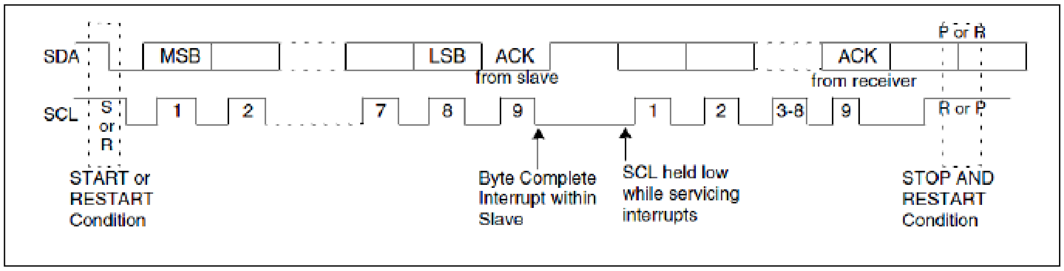
- Protocols Overview
- DMA Controller
- Reset
- Power Management
- Interrupts
- Error Handling
- Programmable SDA Hold Time

28.3.1 Protocols Overview

For more information on the I^{2}C protocols and command formats, refer to the industry I2C specification. Below is a simplified description of I^{2}C bus operation:

- The master generates a START condition, signaling all devices on the bus to listen for data.
- The master 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 master 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 master. These messages are specific to the I^{2}C 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 master with a STOP condition. This frees the bus for the next master to begin communications. When the bus is free, both data and clock lines are high.

Figure 45. Data Transfer on I2C Bus



Combined Formats

The PCH I2C controllers support mixed read and write combined format transactions in both 7-bit and 10-bit addressing modes.

The PCH 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 master, when the controller completes an I2C 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 I2C transfer completes, a STOP is issued and the next transfer is issued following a START condition.

28.3.2 DMA Controller

The I^{2}C controllers 0 to 3 (I2C0 - I2C3) each has an integrated DMA controller. The I2C controller 4 and 5 (I2C4 and I2C5) only implement the I2C 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. The block size is not be 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.

28.3.3 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 I2C peripheral deadlock condition where the reset goes active in the middle of a transaction, the I2C controller must be idle before a reset can be initiated.

28.3.4 Power Management

Device Power Down Support

To power down peripherals connected to PCH I^{2}C 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 I2C bus is powered off (power gated). The PCH 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.

28.3.5 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.

All interrupts are active high and their behavior is level triggered.

28.3.6 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.

28.3.7 Programmable SDA Hold Time

PCH includes a software programmable register to enable dynamic adjustment of the SDA hold time, if needed.

29.0 Host System Management Bus (SMBus) Controller

The PCH provides a System Management Bus (SMBus) 2.0 host controller as well as an SMBus Slave Interface. The PCH is also capable of operating in a mode in which it can communicate with I^{2}C compatible devices.

The host SMBus controller supports up to 100 kHz clock speed.

Table 101. Acronyms

Acronyms	Description
ARP	Address Resolution Protocol
CRC	Cyclic Redundancy Check
PEC	Package Error Checking
SMBus	System Management Bus

Table 102. References

Specification	Location
System Management Bus (SMBus) Specification, Version 2.0	http://www.smbus.org/specs/

29.1 Signal Description

Name	Type	Description
GP_A07/ SMB_CLK	I/OD	**SMBus Clock.** External Pull-up resistor is required.
GP_A08/ SMB_DATA	I/OD	**SMBus Data.** External Pull-up resistor is required.
GP_A09/ SMB_ALERT_N	I/OD	**SMBus Alert**: This signal is used to wake the system or generate SMI#. External Pull-up resistor is required.

29.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SMB_ALERT_N	Pull-down	20k ± 30%	The internal pull-down resistor is disabled after PMC_RSMRST_N de-asserted.

29.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
SMB_DATA	Primary	Undriven	Undriven	Undriven	Undriven
SMB_CLK	Primary	Undriven	Undriven	Undriven	Undriven
SMB_ALERT_N	Primary	Undriven	Undriven	Undriven	OFF
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

29.4 Functional Description

The PCH provides an System Management Bus (SMBus) 2.0 host controller as well as an SMBus Slave Interface.

- **Host Controller:** Provides a mechanism for the processor to initiate communications with SMBus peripherals (slaves). The PCH is also capable of operating in a mode in which it can communicate with I^2C compatible devices.
- **Slave Interface:** Allows an external master to read from or write to the PCH. 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 PCH's internal host controller cannot access the PCH's internal Slave Interface.

29.4.1 Host Controller

The host SMBus controller supports up to 100-KHz clock speed and is clocked by the RTC clock.

The PCH 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 PCH 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 slave 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 8 command protocols of the SMBus interface (Refer 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 Slave 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.

Slave functionality, including the Host Notify protocol, is available on the SMBus pins.

Using the SMB host controller to send commands to the PCH SMB slave 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 PCH 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 Slave 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 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 Slave Address and Device Command Registers are sent. For the Receive Byte command, the Transmit Slave 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 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 Slave 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 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 PCH must write a command to the slave device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The slave 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 Slave 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 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 slave 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 PCH transmits the Transmit Slave 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 section 5.5.6 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

NOTE

For process call command, the value written into bit 0 of the Transmit Slave Address Register needs to be 0.

NOTE

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 slave will not acknowledge (Bit 19 in the sequence).

Block Read/Write

The PCH 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 PCH, 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 PCH 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 slave address and a write condition. After the command code the PCH issues a byte count describing how many more bytes will follow in the message. If a slave 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 Slave 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 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 PCH 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^{2}C mode (I2C_EN bit is set), the PCH will never use the 32-byte buffer for any block commands.

I2C* Read

This command allows the PCH to perform block reads to certain I2C devices, such as serial E^{2}PROMs. The SMBus Block Read supports the 7-bit addressing mode only.

However, this does not allow access to devices using the I2C "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 I2C 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^{2}C Read command, the value written into bit 0 of the Transmit Slave Address Register (SMB I/O register, offset 04h) needs to be 0.

The format that is used for the command is shown in this table.

Table 103. I^{2}C* Block Read

Bit	Description
1	Start
8:2	Slave Address – 7 bits
9	Write
10	Acknowledge from slave
18:11	Send DATA1 register
19	Acknowledge from slave

continued...

Bit	Description
20	Repeated Start
27:21	Slave Address – 7 bits
28	Read
29	Acknowledge from slave
37:30	Data byte 1 from slave – 8 bits
38	Acknowledge
46:39	Data byte 2 from slave – 8 bits
47	Acknowledge
–	Data bytes from slave/Acknowledge
–	Data byte N from slave – 8 bits
–	NOT Acknowledge
–	Stop

The PCH 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 slave 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 master 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 slave 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 slave 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.

NOTE

There is no STOP condition before the repeated START condition, and that a NACK signifies the end of the read transfer.

NOTE

E32B bit in the Auxiliary Control register must be set when using this protocol.

Refer section 5.5.8 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Bus Arbitration

Several masters may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The PCH continuously monitors the SMBDATA line. When the PCH is attempting to drive the bus to a 1 by letting go of the SMBDATA line, and it samples SMBDATA low, then some other master is driving the bus and the PCH will stop transferring data.

If the PCH sees that it has lost arbitration, the condition is called a collision. The PCH will set the BUS_ERR bit in the Host Status Register, and if enabled, generate 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 PCH as an SMBus master would like. They have the capability of stretching the low time of the clock. When the PCH attempts to release the clock (allowing the clock to go high), the clock will remain low for an extended period of time.

The PCH 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 master if it is not ready to send or receive data.

Bus Timeout (PCH as SMBus Master)

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 PCH will discard the cycle and set the DEV_ERR bit. The timeout minimum is 25 ms (800 RTC clocks). The Timeout counter inside the PCH will start after the first bit of data is transferred by the PCH 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 PCH 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.

These tables specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and Slave 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 104. Enabling 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 (Slave 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	Slave SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 105. Enabling SMBus Slave 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
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS).
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave 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 106. Enabling for the Host Notify Command

HOST_NOTIFY_INTREN (Slave Control I/O Register, Offset 11h, Bit 0)	SMB_SMI_EN (Host Config Register, D31:F4:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Slave 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	Slave SMI# generated (SMBUS_SMI_STS)

SMBus CRC Generation and Checking

If the AAC bit is set in the Auxiliary Control register, the PCH 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.

29.4.2 SMBus Slave Interface

The PCH SMBus Slave interface is accessed using the SMBus. The SMBus slave logic will not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol device. The slave interface allows the PCH to decode cycles, and allows an external microcontroller to perform specific actions.

Key features and capabilities include:

- **Supports decode of three types of messages:** Byte Write, Byte Read, and Host Notify.
- **Receive Slave Address register:** This is the address that the PCH decodes. A default value is provided so that the slave interface can be used without the processor having to program this register.
- Receive Slave Data register in the SMBus I/O space that includes the data written by the external microcontroller.
- Registers that the external microcontroller can read to get the state of the PCH.
  - Status bits to indicate that the SMBus slave logic caused an interrupt or SMI# Bit 0 of the Slave Status Register for the Host Notify command
  - Bit 16 of the SMI Status Register for all others

    The external microcontroller should not attempt to access the PCH SMBus slave logic until either:

    - 800 milliseconds after both: RTCRST# is high and PMC_RSMRST_N is high, OR
    - The PLTRST# de-asserts

If a master leaves the clock and data bits of the SMBus interface at 1 for 50 µs or more in the middle of a cycle, the PCH slave logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the slave logic.

Format of Slave Write Cycle

The external master performs Byte Write commands to the PCH SMBus Slave 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.

This table has the values associated with the registers.

Table 107. Slave Write Registers

Register	Function
0	Command Register. Refer to table below for valid values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–7	Reserved

continued...

Register	Function
8	Reserved
9–FFh	Reserved
Note: The external microcontroller 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 PCH 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 PCH will not attempt to cover this race condition (that is, unpredictable results in this case).	

Table 108. 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 Powerbutton 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 PCH 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 PMC_RSMRST_N signal.
6	**WD RELOAD:** Reload watchdog timer.
7	Reserved
8	**SMLINK_SLV_SMI.** When the PCH 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 S3–S5 states, the PCH 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 master performs Byte Read commands to the PCH SMBus Slave 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 109. Slave Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Microcontroller	
2–8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
9	Write	External Microcontroller	Always 0

continued...

Bit	Description	Driven By	Comment
10	ACK	PCH	
11–18	Command code – 8 bits	External Microcontroller	Indicates which register is being accessed. Refer to table below for a list of implemented registers.
19	ACK	PCH	
20	Repeated Start	External Microcontroller	
21–27	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
28	Read	External Microcontroller	Always 1
29	ACK	PCH	
30–37	Data Byte	PCH	Value depends on register being accessed. Refer to table below for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 110. Slave Read Registers Data Values

Register	Bits	Description
0	7:0	Reserved
1	2:0	**System Power State** 000 = S0 011 = S3 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 PCH will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect**. Reserved
	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.
	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	**FWH bad bit.** This bit will be 1 to indicate that the FWH read returned FFh, which indicates that it is probably blank.

continued...

Register	Bits	Description
	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 PCH_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 PCH detects that the address matches the value in the Receive Slave 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 PCH’s Slave Address, the PCH 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 PCH’s Receive Slave Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Slave Read cycle.

Slave Read of RTC Time Bytes

The PCH SMBus slave interface allows external SMBus master to read the internal RTC’s time byte registers.

The RTC time bytes are internally latched by the PCH's hardware whenever RTC time is not changing and SMBus is idle. This ensures that the time byte delivered to the slave 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 PCH SMBus slave interface only supports Byte Read operation. The external SMBus master will read the RTC time bytes one after another. It is 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 master 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 PCH 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 PCH 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 master 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 following table shows the Host Notify format.

Table 111. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Master	
8:2	SMB Host Address – 7 bits	External Master	Always 0001_000
9	Write	External Master	Always 0
10	ACK (or NACK)	PCH	PCH NACKs if HOST_NOTIFY_STS is 1
17:11	Device Address – 7 bits	External Master	Indicates the address of the master; loaded into the Notify Device Address Register
18	Unused – Always 0	External Master	7-bit-only address; this bit is inserted to complete the byte
19	ACK	PCH	
27:20	Data Byte Low – 8 bits	External Master	Loaded into the Notify Data Low Byte Register
28	ACK	PCH	

continued...

Bit	Description	Driven By	Comment
36:29	Data Byte High – 8 bits	External Master	Loaded into the Notify Data High Byte Register
37	ACK	PCH	
38	Stop	External Master	

Format of Read Command

The external master performs Byte Read commands to the PCH SMBus Slave 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 112. Slave Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Microcontroller	
2–8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
9	Write	External Microcontroller	Always 0
10	ACK	PCH	
11–18	Command code – 8 bits	External Microcontroller	Indicates which register is being accessed. Refer to table Table 110 on page 232 for a list of implemented registers.
19	ACK	PCH	
20	Repeated Start	External Microcontroller	
21–27	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
28	Read	External Microcontroller	Always 1
29	ACK	PCH	
30–37	Data Byte	PCH	Value depends on register being accessed. Refer to table Table 110 on page 232 for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 113. Data Values for Slave 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 011 = S3 100 = S4 101 = S5 Others = Reserved
	7:3	Reserved

continued...

Register	Bits	Description
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 PCH 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.Reserved
	1	**Temperature Event.** 1 = Temperature Event occurred. This bit will be set if the PCH'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 SLP_S3# pin is de-asserted and PCH_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 114. Enabling SMBus Slave 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
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS)
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave 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

29.5 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.

30.0 System Management Interface and SMLink

The PCH provides one SMLink interface (SMLink0). The interface is intended for USB Type C PD management. Refer to System Management on page 239 for more detail.

30.1 Signal Description

Name	Type	Description
SML_DATA0	I/OD	System Management Link 0 Data External Pull-up resistor required.
SML_CLK0	I/OD	System Management Link 0 Clock External Pull-up resistor required.
PMC_ALERT_N	I/OD	For USB Type-C* PD Controller External pull-up resistor required

30.2 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
SML_DATA0	Primary	Undriven	Undriven	Undriven	Undriven
SML_CLK0	Primary	Undriven	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.					

31.0 System Management

The PCH provides various functions to make a system easier to manage and to lower the Total Cost of Ownership (TCO) of the system. Features and functions can be augmented using external A/D converters and GPIOs, as well as an external micro controller.

The following features and functions are supported by the PCH:

- First timer timeout to generate SMI# after programmable time
  - The first timer timeout causes an SMI#, allowing SMM-based recovery from OS lock up
- Second hard-coded timer timeout to generate reboot:
  - This second timer is used only after the 1st timeout occurs
  - The second timeout allows for automatic system reset and reboot if a HW error is detected
  - Option to prevent reset the second timeout via HW strap
- Various Error detection (such as ECC Errors) indicated by host controller:
  - Can generate SMI#, SCI, SERR, SMI, or TCO interrupt

31.1 Theory of Operation

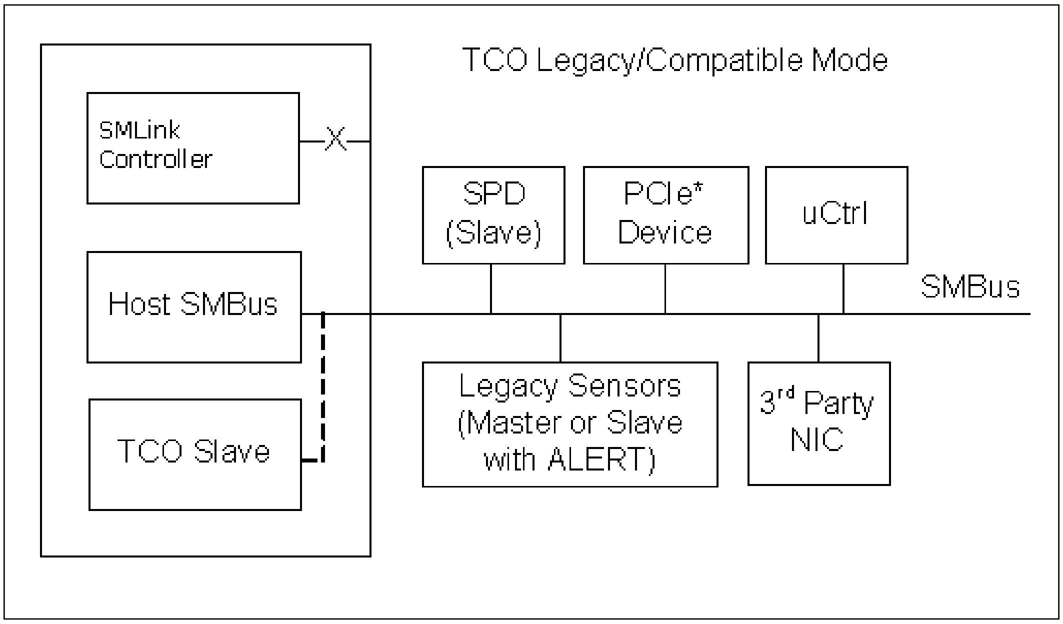
The System Management functions are designed to allow the system to diagnose failing subsystems. The intent of this logic is that some of the system management functionality can be provided without the aid of an external microcontroller.

31.1.1 TCO Modes

TCO Compatible Mode

In TCO Legacy/Compatible mode, only the host SMBus is used. The TCO Slave is connected to the host SMBus internally by default.

Figure 46. TCO Compatible Mode SMBus Configuration



In TCO Legacy/Compatible mode the PCH can function directly with an external LAN controller or equivalent external LAN controller to report messages to a network management console without the aid of the system processor. This is crucial in cases where the processor is malfunctioning or cannot function due to being in a low-power state. Below table includes a list of events that will report messages to the network management console.

Table 115. Event Transitions that Cause Messages

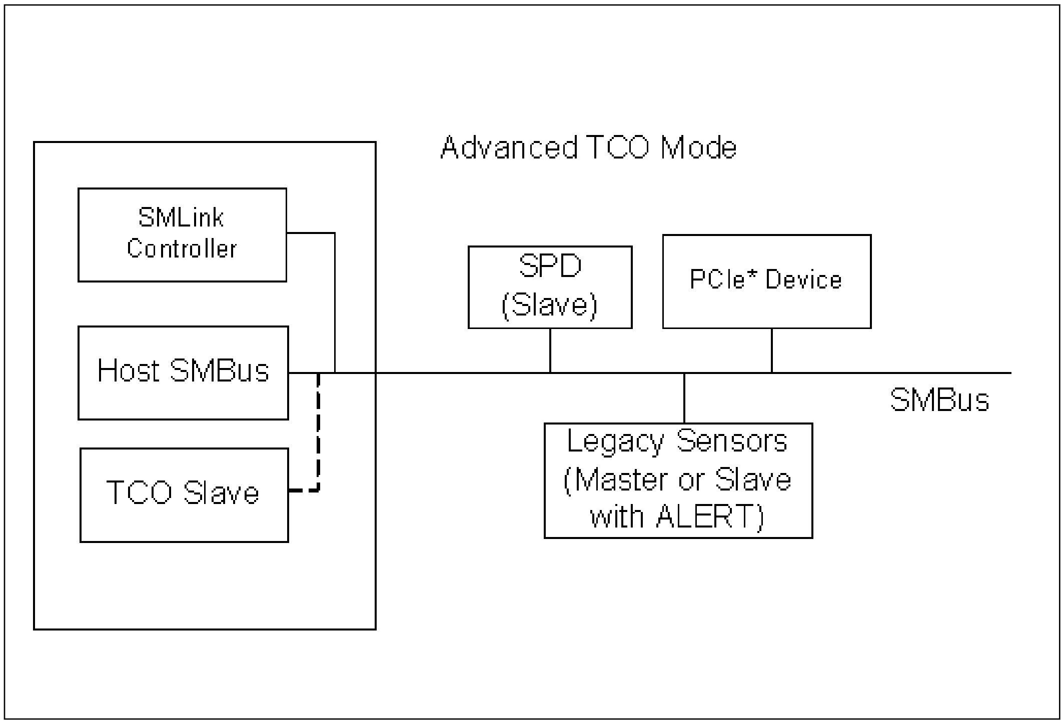
Event	Assertion?	Deassertion?	Comments
Watchdog Timer Expired	Yes	NA	“Hung S0” state entered
SMBALERT# pin	Yes	Yes	Must be in “Hung S0” state
BATLOW#	Yes	Yes	Must be in “Hung S0” state
CPU_PWR_FLR	Yes	No	“Hung S0” state entered

Advanced TCO Mode

The PCH supports the Advanced TCO mode in which SMLink are used in addition to the host SMBus.

In advanced TCO mode, the TCO slave can either be connected to the host SMBus or the SMLink0.

Figure 47. Advanced TCO Mode



32.0 Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers

The PCH 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 Mbits/s Auto Flow Control mode as specified in the 16750 standard
- 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 116. Acronyms

Acronyms	Description
DMA	Direct Memory Access
UART	Universal Asynchronous Receiver/Transmitter

32.1 Signal Description

Name	Type	Description
UART0_RXD	I	UART 0 Receive Data
UART0_TXD	O	UART 0 Transmit Data
UART0_RTS_N	O	UART 0 Request to Send
UART0_CTS_N	I	UART 0 Clear to Send
UART0A_RXD	I	Alternate muxing of UART 0 Receive Data
UART0A_TXD	O	Alternate muxing of UART 0 Transmit Data
UART0A_RTS_N	O	Alternate muxing of UART 0 Request to Send
UART0A_CTS_N	I	Alternate muxing of UART 0 Clear to Send
UART1_RXD	I	UART 1 Receive Data
UART1_TXD	O	UART 1 Transmit Data
UART1_RTS_N	O	UART 1 Request to Send
UART1_CTS_N	I	UART 1 Clear to Send
UART2_RXD	I	UART 2 Receive Data
UART2_TXD	O	UART 2 Transmit Data
UART2_RTS_N	O	UART 2 Request to Send
UART2_CTS_N	I	UART 2 Clear to Send

32.2 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S3/S4/S5	Deep Sx
UART[2:0]_RXD	Primary	Undriven	Undriven	Undriven	OFF
UART[2:0]_TXD	Primary	Undriven	Undriven	Undriven	OFF
UART[2:0]_RTS_N	Primary	Undriven	Undriven	Undriven	OFF
UART[2:0]_CTS_N	Primary	Undriven	Undriven	Undriven	OFF

Note: 1. Reset reference for primary well pins is PMC_RSMRST_N.

32.3 Functional Description

This section contains information about the following:

- UART Serial (RS-232) Protocols Overview

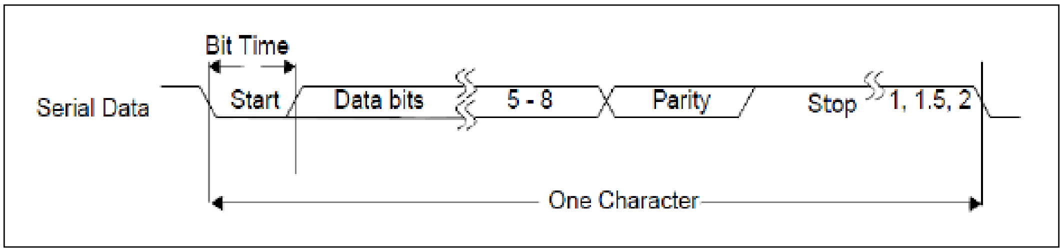
- 16550 8-bit Addressing - Debug Driver Compatibility
- DMA Controller
- Reset
- Power Management
- Interrupts
- Error Handling

32.3.1 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 48. UART Serial Protocol



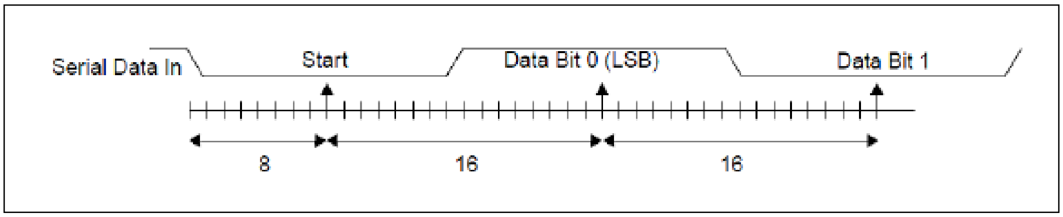
The UART Host Controller Line Control Register (LCR) is used to control the serial character characteristics. The individual bits of the data word are sent after the Start bit, starting with the least significant bit (LSB). These are followed by the optional parity bit, followed by the Stop bit(s), which 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 49. UART Receiver Serial Data Sample Points



32.3.2 16550 8-bit Addressing - Debug Driver Compatibility

NOTE

The PCH 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 (byte) 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.

32.3.3 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 incorporate 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. the block size is not be 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.

32.3.4 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.

32.3.5 Power Management

Device Power Down Support

In order to power down peripherals connected to PCH UART 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 PCH 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.

32.3.6 Interrupts

UART 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 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.

All interrupts are active high and their behavior is level interrupt.

32.3.7 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.

33.0 Testability

This chapter provides information regarding testability of on the following topics:

Intel® Processor Trace

Intel® Processor Trace (Intel® PT) is a tracing capability added to Intel® Architecture, for use in software debug and profiling. Intel® PT provides the capability for precise software control flow and timing information, with limited impact to software execution. This provides enhanced ability to debug software crashes, hangs, or other anomalies, as well as responsiveness and short-duration performance issues. Refer to the Intel® 64 Architectures Software Developer’s Manual, for more information: https://www.intel.com/content/www/us/en/products/processors.html/manuals.

NOTE

Intel® Processor Trace uses CFG bus.

JTAG

This provides information regarding the testability signals that provides access to JTAG, run control, system control, and observation resources. JTAG (TAP) ports are compatible with the IEEE Standard Test Access Port and Boundary Scan Architecture 1149.1 and 1149.6 Specification, as detailed per device in each BSDL file. JTAG Pin definitions are from IEEE Standard Test Access Port and Boundary Scan. Architecture (IEEE Std. 1149.1-2001).

Intel® Trace Hub

Intel® Trace Hub is a debug architecture that unifies hardware and software system visibility. Intel® Trace Hub is not merely intended for hardware debug or software debug, but full system debug. This includes debugging hardware and software as they interact and produce complex system behavior. Intel® Trace Hub defines new features and also leverages some existing debug technologies to provide a complete framework for hardware and software co-debug, software development and tuning, as well as overall system performance optimization.

Intel® Trace Hub is a set of silicon features with supported software API. The primary purpose is to collect trace data from different sources in the system and combine them into a single output stream with time-correlated to each other. Intel® Trace Hub uses common hardware interface for collecting time-correlated system traces through standard destinations. Intel® Trace Hub adopts industry standard (MIPI* STPv2) debug methodology for system debug and software development.

There are multiple destinations to receive the trace data from Intel® Trace Hub:

- Direct Connect Interface (DCI)
  - — OOB Hosting DCI
  - — USB 2.0 and USB 3.2 hosting DCI.DBC
- System Memory

There are multiple trace sources planned to be supported in the platform:

- BIOS
- Intel® CSE
- AET (Architecture Event Trace)
- Power Management Event Trace
- Windows* ETW (for driver or application)

Table 117. Acronyms

Acronyms	Description
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
I/OD	Input/Output Open Drain
JTAG	Joint Test Action Group
DCI	Direct Connect Interface
BSDL	Boundary Scan Description Language
DbC	Debug Class Devices

Table 118. References

Specification	Location
IEEE Standard Test Access Port and Boundary Scan Architecture	http://standards.ieee.org/findstds/standard/1149.1-2013.html

33.1 Intel® Trace Hub (Intel® TH)

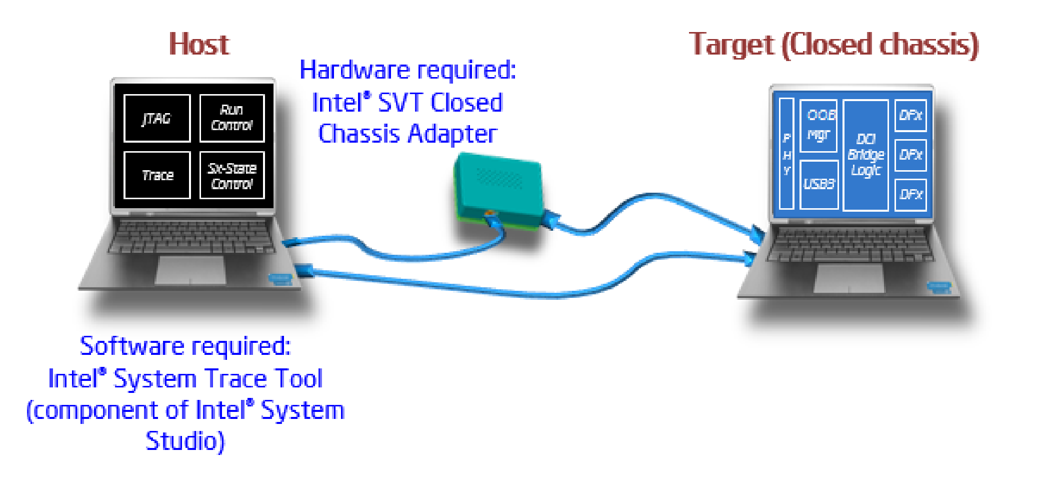
Table 119. JTAG, DBG_PMODE and CFG Testability Signal

Signal Name	Type[1]	Description
CPU_JTAG_TCK	IN	Test Clock Input (TCK): The test clock input provides the clock for the JTAG test logic
PCH_JTAG_X	IN	Pin used to support Merged Debug Port topology
CPU_JTAG_TMS PCH_JTAG_TMS	IN	Test Mode Select (TMS): The signal is decoded by the Test Access Port (TAP) controller to control test operations
CPU_JTAG_TDI PCH_JTAG_TDI	IN	Test Data Input (TDI): Serial test instructions and data are received by the test logic at TDI
CPU_JTAG_TDO PCH_JTAG_TDO	OUT	Test Data Output (TDO): TDO is the serial output for test instructions and data from the test logic defined in this standard
CPU_JTAG_TRST_N PCH_JTAG_TRST_N	IN	Test Reset (active low)
DBG_PMODE[2]	IN/OUT	Debug Power Mode Indicator. Signal is used to transmit Compute Die and PCH power/reset information to the debug tool
CFG[00:15][3]	IN/OUT	CFG (Parallel Trace Interface) signals are used for Compute Die Tracing

Notes: 1. Directions are specified at Processor
2. DBG_PMODE (HOOK 6) part of Miscellaneous Signal
3. CFG[00:15] part from Trace Signal

33.1.1 Platform Setup

Figure 50. Platform Setup with Intel® Trace Hub



33.2 Direct Connect Interface (DCI)

Direct Connect Interface (DCI) is a new debug transport technology to enable closed chassis debug through any of USB 3.2 ports out from Intel silicon. Some bridging logic is embedded in the silicon to "bridge" the gap between standard I/O ports and the debug interfaces including JTAG, probe mode, hooks, trace infrastructure, and so on. To control the operation of this embedded logic, a DCI packet based protocol is invented which controls and data can be sent or received. This protocol can operate over a few different physical transport paths to the target which known as "hosting interfaces".

NOTE

DCI and USB 3.2 based debugger (kernel level debugger) are mutually exclusive.

There are two types of DCI hosting interfaces in the platform:

- OOB Hosting DCI
- USB 2.0 and USB 3.2 Hosting DCI.DBC

Supported capabilities in DCI are:

- Closed Chassis Debug at S0 and Sx State
- JTAG Access and Run Control (Probe Mode)
- System Tracing with Intel® Trace Hub

Debug host software that support DCI are:

- Intel® ITP II Platform Debug Toolkit (PDT)
- Intel® System Studio (ISS)

33.2.1 Out Of Band (OOB) Hosting DCI

OOB was developed to provide an alternate path to convey controls and data to or from the EXI/DCI by connecting physically to the target through a USB 3.2 port. OOB provides an alternate side band path around the USB 3.2 controller, so that the embedded logic can be accessed, even when the USB 3.2 controller is not alive (such as in low power states) or is malfunctioning. This path does not rely on USB 3.2 protocol, link layer, or physical layer, because the xHCI functions are generally not available in such conditions. Instead, this path relies on a special adapter that was developed by Intel called the Intel® SVT Closed Chassis Adapter (CCA). It is a simple data transformation device. This adapter generates a OOB signaling protocol operating at up to 400 MHz and serializes data flowing through it. This adapter works together with debug host software and the embedded logic, contain a back-pressure scheme that makes both sides tolerant of overflow and starvation conditions, which is equivalent of USB 3.2 link layer. This path also uses native DCI packet protocol instead of USB 3.2 protocol. DCI.OOB - slower speed, CCA box needed. But survives S0ix and Sx states. Provides early boot access. Cannot tolerate re-driver circuits in its path.

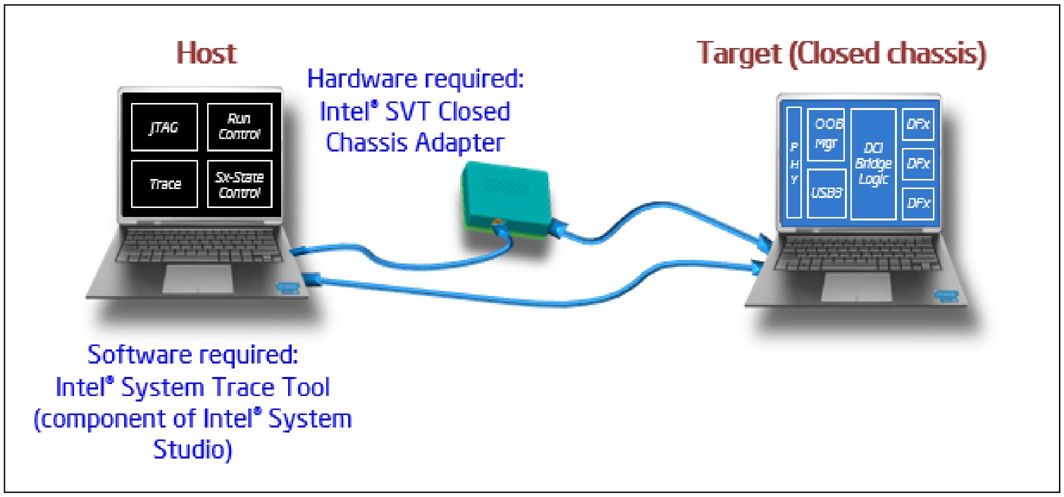
Intel® SVT CCA (MM#:921521) can be purchased through Intel® Design-In Tools Store at https://designintools.intel.com/product_p/itpxdpsvt.htm.

33.2.2 USB 3.2 Hosting DCI.DBC

It relies on Debug Class Devices (DbC) which is comprised of a set of logic that is bolted to the side of the xHCI host controller and enable the target to act the role of a USB 3.2 device for debug purpose. This path uses the USB 3.2 packet protocol layer, USB 3.2 link layer flow control and USB 3.2 physical layer at 5 GHz. DCI.DBC - fast speed. USB 3.2 only works in S0. USB 2.0 survives S0ix and Sx states and provides early boot access.

33.2.3 Platform Setup

Figure 51. Platform Setup with DCI Connection



34.0 SoC Pin Location

Table 120. SoC Pin List

Ball#	External Name
BE3	XTAL_OUT
BE1	XTAL_IN
BU47	PMC_WAKE_N
A10	VSS
A16	VSS
A21	VSS
A23	VSS
A27	VSS
A29	VSS
A32	VSS
A34	VSS
A37	VSS
A4	VSS
A40	VSS
A43	VSS
A48	VSS
A49	VSS
A5	VSS
AA47	VSS
AA50	VSS
AA52	VSS
AB1	VSS
AB10	VSS
AB15	VSS
AB17	VSS
AB18	VSS
AB20	VSS
AB21	VSS
AB23	VSS
AB25	VSS
AB33	VSS
AB38	VSS
AB6	VSS
AC20	VSS
AC21	VSS
AC23	VSS
AC25	VSS
AC33	VSS
AC47	VSS
AD10	VSS
AD15	VSS
AD38	VSS
AD52	VSS
AD6	VSS
AE17	VSS
AE18	VSS
AE33	VSS
AF1	VSS
AF39	VSS
AF41	VSS
AF42	VSS
AF44	VSS
AF45	VSS
AF47	VSS
AF50	VSS
AF52	VSS
AG13	VSS
AG15	VSS
AG17	VSS
AG18	VSS

continued...

Ball#	External Name
AG33	VSS
AG34	VSS
AG9	VSS
AH17	VSS
AH18	VSS
AH33	VSS
AH38	VSS
AH47	VSS
AJ10	VSS
AJ13	VSS
AJ52	VSS
AJ6	VSS
AK1	VSS
AK18	VSS
AK38	VSS
AL47	VSS
AL50	VSS
AL52	VSS
AM13	VSS
AM18	VSS
AM33	VSS
AM39	VSS
AM41	VSS
AM42	VSS
AM44	VSS
AM45	VSS
AM6	VSS
AN1	VSS
AP18	VSS
AP38	VSS
AP47	VSS
AP52	VSS
AR13	VSS
AR18	VSS
AR33	VSS
AR9	VSS
AT38	VSS
AU1	VSS
AU10	VSS
AU18	VSS
AU3	VSS
AU47	VSS
AU50	VSS
AU52	VSS
AU6	VSS
AV39	VSS
AV41	VSS
AV42	VSS
AV44	VSS
AV45	VSS
AW13	VSS
AW33	VSS
AW47	VSS
AW9	VSS
AY23	VSS
AY33	VSS
AY38	VSS
AY52	VSS
B12	VSS
B4	VSS
B5	VSS
BA10	VSS
BA15	VSS
BA6	VSS
BB17	VSS
BB18	VSS
BB20	VSS
BB21	VSS
BB23	VSS
BB33	VSS
BB38	VSS
BB39	VSS

continued...

Ball#	External Name
BB47	VSS
BB50	VSS
BB52	VSS
BC1	VSS
BC9	VSS
BD2	VSS
BD21	VSS
BD23	VSS
BD39	VSS
BD4	VSS
BD41	VSS
BD42	VSS
BD44	VSS
BD45	VSS
BE21	VSS
BE23	VSS
BE47	VSS
BE52	VSS
BF13	VSS
BF15	VSS
BF2	VSS
BF38	VSS
BF4	VSS
BF9	VSS
BG21	VSS
BG23	VSS
BG47	VSS
BG50	VSS
BG52	VSS
BH38	VSS
BJ1	VSS
BJ21	VSS
BJ23	VSS
BJ3	VSS
BJ33	VSS
BJ9	VSS
BK21	VSS
BK23	VSS
BK39	VSS
BK41	VSS
BK42	VSS
BK44	VSS
BK45	VSS
BK47	VSS
BK52	VSS
BL10	VSS
BL15	VSS
BL6	VSS
BM1	VSS
BM17	VSS
BM18	VSS
BM20	VSS
BM21	VSS
BM23	VSS
BM3	VSS
BM33	VSS
BM38	VSS
BN47	VSS
BN50	VSS
BN52	VSS
BP17	VSS
BP21	VSS
BP23	VSS
BP33	VSS
BP38	VSS
BP48	VSS
BP9	VSS
BR47	VSS
BT1	VSS
BT17	VSS
BT21	VSS
BT23	VSS

continued...

Ball#	External Name
BT25	VSS
BT26	VSS
BT28	VSS
BT3	VSS
BT30	VSS
BT31	VSS
BT33	VSS
BT39	VSS
BT41	VSS
BT42	VSS
BT44	VSS
BT45	VSS
BT52	VSS
BU10	VSS
BU17	VSS
BU21	VSS
BU23	VSS
BU25	VSS
BU26	VSS
BU28	VSS
BU30	VSS
BU31	VSS
BU33	VSS
BU34	VSS
BU6	VSS
BV47	VSS
BV50	VSS
BV52	VSS
BW13	VSS
BW17	VSS
BW2	VSS
BW21	VSS
BW4	VSS
BW9	VSS
BY42	VSS
C13	VSS

continued...

Ball#	External Name
C16	VSS
C21	VSS
C23	VSS
C29	VSS
C34	VSS
C40	VSS
C50	VSS
CA17	VSS
CA21	VSS
CA47	VSS
CA52	VSS
CA9	VSS
CB1	VSS
CB26	VSS
CB3	VSS
CB32	VSS
CB34	VSS
CB36	VSS
CC32	VSS
CC39	VSS
CC44	VSS
CC45	VSS
CC47	VSS
CC50	VSS
CC52	VSS
CD38	VSS
CD6	VSS
CD7	VSS
CE21	VSS
CE24	VSS
CE30	VSS
CE47	VSS
CE9	VSS
CF13	VSS
CF17	VSS
CF36	VSS

continued...

Ball#	External Name
CF42	VSS
CF52	VSS
CG1	VSS
CH43	VSS
CJ15	VSS
CJ21	VSS
CJ23	VSS
CJ28	VSS
CJ32	VSS
CJ34	VSS
CJ38	VSS
CJ4	VSS
CJ52	VSS
CJ7	VSS
CK43	VSS
CL15	VSS
CL18	VSS
CL21	VSS
CL24	VSS
CL26	VSS
CL29	VSS
CL32	VSS
CL34	VSS
CL37	VSS
CL40	VSS
CL47	VSS
CM1	VSS
CM10	VSS
CM52	VSS
CN47	VSS
CN6	VSS
CP12	VSS
CP15	VSS
CP21	VSS
CP26	VSS
CP32	VSS

continued...

Ball#	External Name
CP37	VSS
CR1	VSS
CR50	VSS
CT15	VSS
CT18	VSS
CT21	VSS
CT23	VSS
CT26	VSS
CT28	VSS
CT32	VSS
CT34	VSS
CT37	VSS
CT39	VSS
CT4	VSS
CT42	VSS
CT45	VSS
CT48	VSS
CT5	VSS
CT51	VSS
CT8	VSS
D1	VSS
D16	VSS
D21	VSS
D52	VSS
D6	VSS
E52	VSS
E7	VSS
F12	VSS
F13	VSS
F23	VSS
F26	VSS
F29	VSS
F31	VSS
F34	VSS
F37	VSS
F4	VSS

continued...

Ball#	External Name
F40	VSS
F43	VSS
F47	VSS
G10	VSS
G5	VSS
H1	VSS
H10	VSS
H16	VSS
H20	VSS
H26	VSS
H32	VSS
H38	VSS
H39	VSS
H41	VSS
H42	VSS
H44	VSS
H45	VSS
H46	VSS
H8	VSS
J10	VSS
J26	VSS
J32	VSS
J47	VSS
J52	VSS
K38	VSS
K47	VSS
K5	VSS
L10	VSS
L16	VSS
L20	VSS
L26	VSS
L32	VSS
L6	VSS
L7	VSS
L8	VSS
M26	VSS
M32	VSS
M38	VSS
M47	VSS
M52	VSS
N11	VSS
N3	VSS
P1	VSS
P16	VSS
P20	VSS
P26	VSS
P32	VSS
P39	VSS
P41	VSS
P42	VSS
P44	VSS
P45	VSS
R13	VSS
R26	VSS
R28	VSS
R30	VSS
R32	VSS
R34	VSS
R36	VSS
R47	VSS
R50	VSS
R52	VSS
R9	VSS
T38	VSS
U12	VSS
U13	VSS
U20	VSS
U21	VSS
U23	VSS
U25	VSS
U26	VSS
U6	VSS

continued...

Ball#	External Name
V1	VSS
V17	VSS
V18	VSS
V20	VSS
V21	VSS
V26	VSS
V28	VSS
V3	VSS
V30	VSS
V33	VSS
V38	VSS
V47	VSS
V52	VSS
W10	VSS
W13	VSS
W9	VSS
Y17	VSS
Y18	VSS
Y20	VSS
Y21	VSS
Y23	VSS
Y25	VSS
Y33	VSS
Y39	VSS
Y41	VSS
Y42	VSS
Y44	VSS
Y45	VSS
AC2	SVID_DATA
AA2	SVID_CLK
AD3	SVID_ALERT_N
CC17	eDP_VDDEN
BK34	VCCSTG_OUT
AW3	VCCSTG
AW1	VCCSTG
AP4	VCCST_PWRGD

continued...

Ball#	External Name
AP2	VCCST_OVERRIDE
AY2	VCCST
BA1	VCCST
BA3	VCCST
BV38	VCCPLL_OC
BV39	VCCPLL_OC
AY4	VCCPLL
CA30	VCCSPI
CL49	VCCRTC
CA36	VCCPRIM_1P05
BW36	VCCPRIM_1P05
BW31	VCCPRIM_3P3
BW33	VCCPRIM_3P3
BW34	VCCPRIM_3P3
CA31	VCCPRIM_3P3
CA33	VCCPRIM_3P3
CA34	VCCPRIM_3P3
BW23	VCCPRIM_1P8
BW25	VCCPRIM_1P8
BW26	VCCPRIM_1P8
CA23	VCCPRIM_1P8
CA25	VCCPRIM_1P8
CA26	VCCPRIM_1P8
BW28	VCCPGPPR
BW30	VCCPRIM_3P3
CA28	VCCPRIM_1P8
CN49	VCCLDOSTD_0P85
AJ12	VCCIO_SENSE
AJ15	VCCIO_EXT
AM15	VCCIO_EXT
AR15	VCCIO_EXT
AU15	VCCIO_EXT
AW15	VCCIO_EXT
AU8	VCCIN_AUX_VSSSENSE
AU9	VCCIN_AUX_SENSE
J24	VCCIN_VSS_SENSE

continued...

Ball#	External Name
H24	VCCIN_SENSE
BD17	VCCIN_AUX
BD18	VCCIN_AUX
BD20	VCCIN_AUX
BE17	VCCIN_AUX
BE18	VCCIN_AUX
BE20	VCCIN_AUX
BG17	VCCIN_AUX
BG18	VCCIN_AUX
BG20	VCCIN_AUX
BJ17	VCCIN_AUX
BJ18	VCCIN_AUX
BJ20	VCCIN_AUX
BK17	VCCIN_AUX
BK18	VCCIN_AUX
BK20	VCCIN_AUX
A13	VCCIN
A18	VCCIN
B14	VCCIN
B17	VCCIN
B20	VCCIN
B22	VCCIN
C18	VCCIN
D14	VCCIN
D17	VCCIN
D20	VCCIN
D22	VCCIN
E14	VCCIN
E17	VCCIN
E19	VCCIN
E22	VCCIN
F14	VCCIN
F15	VCCIN
F17	VCCIN
F18	VCCIN
F19	VCCIN
F21	VCCIN
F22	VCCIN
H12	VCCIN
H14	VCCIN
H18	VCCIN
H22	VCCIN
J12	VCCIN
J14	VCCIN
J16	VCCIN
J18	VCCIN
J20	VCCIN
J22	VCCIN
L12	VCCIN
L14	VCCIN
L18	VCCIN
L22	VCCIN
L24	VCCIN
M12	VCCIN
M14	VCCIN
M16	VCCIN
M18	VCCIN
M20	VCCIN
M22	VCCIN
M24	VCCIN
P14	VCCIN
P18	VCCIN
P22	VCCIN
P24	VCCIN
R16	VCCIN
R18	VCCIN
R20	VCCIN
R22	VCCIN
R24	VCCIN
U17	VCCIN
U18	VCCIN
BG34	VCCSTG_OUT

continued...

Ball#	External Name
BJ34	VCCSTG_OUT
BE34	VCCSTG_OUT
CK48	VCCDSW_3P3
CM48	VCCDPHY_1P24
AB34	VDDQ
AB36	VDDQ
AC34	VDDQ
AC36	VDDQ
AE34	VDDQ
AE36	VDDQ
AF38	VDDQ
AG36	VDDQ
AH34	VDDQ
AH36	VDDQ
AK34	VDDQ
AK36	VDDQ
AM34	VDDQ
AM36	VDDQ
AM38	VDDQ
AP36	VDDQ
AR34	VDDQ
AR36	VDDQ
AU34	VDDQ
AU36	VDDQ
AV38	VDDQ
AW34	VDDQ
AW36	VDDQ
AY34	VDDQ
AY36	VDDQ
BB36	VDDQ
BD36	VDDQ
BD38	VDDQ
BE36	VDDQ
BG36	VDDQ
BJ36	VDDQ
BK36	VDDQ

continued...

Ball#	External Name
BK38	VDDQ
BM36	VDDQ
BP36	VDDQ
P38	VDDQ
U34	VDDQ
U36	VDDQ
V34	VDDQ
V36	VDDQ
Y34	VDDQ
Y36	VDDQ
Y38	VDDQ
CN1	VCCA_CLKLDO_1P8
CN3	VCCA_CLKLDO_1P8
AV2	VCC1P8A
AV4	VCC1P8A
BB4	VCC1P05_OUT_PLL
BU36	VCCPRIM_1P05
BB2	VCC1P05_OUT
BC3	VCC1P05_OUT
BP20	VCC_VNNEXT_1P05
BT20	VCC_VNNEXT_1P05
BU20	VCC_VNNEXT_1P05
BW20	VCC_VNNEXT_1P05
CA20	VCC_VNNEXT_1P05
BP18	VCC_V1P05EXT_1P05
BT18	VCC_V1P05EXT_1P05
BU18	VCC_V1P05EXT_1P05
BW18	VCC_V1P05EXT_1P05
CA18	VCC_V1P05EXT_1P05
BP13	RSVD_TP
BP15	RSVD_TP
BU8	USB31_2_TXP
BU9	USB31_2_TXN
CA2	USB31_2_RXP
CA4	USB31_2_RXN
BU12	USB31_1_TXP

continued...

Ball#	External Name
BU13	USB31_1_TXN
BY1	USB31_1_RXP
BY3	USB31_1_RXN
CC4	RSVD_TP
BW8	USB2P_8
CG3	USB2P_7
BW10	USB2P_6
CA10	USB2P_5
CD3	USB2P_4
CE2	USB2P_3
CA6	USB2P_2
CF3	USB2P_1
BW6	USB2N_8
CG4	USB2N_7
BW12	USB2N_6
CA12	USB2N_5
CD1	USB2N_4
CE4	USB2N_3
CA8	USB2N_2
CF1	USB2N_1
CC7	USB_VBUSSENSE
CC6	USB_ID
CC2	USB2_RCOMP
AT4	RSVD
AR3	THRMTRIP_N
AP17	RSVD
U28	RSVD
BB34	RSVD
AM17	RSVD
U30	RSVD
BD34	RSVD
CA13	RSVD
CA15	RSVD
CF9	PMC_SYS_RESET_N
CC9	PMC_SYS_PWROK
BR48	SRTCRST_N
CH32	FSPI_CS2_N
CC34	FSPI_MOSI_IO0
CJ36	FSPI_MISO_IO1
CF32	FSPI_IO3
CE32	FSPI_IO2
CF34	FSPI_CS1_N
CE34	FSPI_CS0_N
CH34	FSPI_CLK
CR25	SNDW_RCOMP
BU51	PMC_SLP_SUS_N
CM50	SD3_RCOMP
BJ15	RSVD_TP
BJ13	RSVD_TP
BF8	SATA_1_TXP/USB30_4_TXP/ PCIE_8_TXP
BF6	SATA_1_TXN/USB30_4_TXN/ PCIE_8_TXN
BJ4	SATA_1_RXP/USB30_4_RXP/ PCIE_8_RXP
BJ2	SATA_1_RXN/USB30_4_RXN/ PCIE_8_RXN
BF12	SATA_0_TXP/PCIE_7_TXP
BF10	SATA_0_TXN/PCIE_7_TXN
BK3	SATA_0_RXP/PCIE_7_RXP
BK1	SATA_0_RXN/PCIE_7_RXN
BP47	RTCRST_N
CE48	RTC_X2
CE49	RTC_X1
BU49	PMC_RSMRST_N
AC4	PROCHOT_N
Y1	JTAG_PREQ_N
Y3	JTAG_PRDY_N
AW18	RSVD_TP
AY21	RSVD
AY20	RSVD
AM4	PECI
BH2	PCIE_RCOMPP
BH4	PCIE_RCOMPN

continued...

Ball#	External Name
BP8	PCIE_6_TXP/USB30_1_TXP
BP6	PCIE_6_TXN/USB30_1_TXN
BV1	PCIE_6_RXP/USB30_1_RXP
BV3	PCIE_6_RXN/USB30_1_RXN
BP10	PCIE_5_TXP/USB30_2_TXP
BP12	PCIE_5_TXN/USB30_2_TXN
BU4	PCIE_5_RXP/USB30_2_RXP
BU2	PCIE_5_RXN/USB30_2_RXN
BL9	PCIE_4_TXP/USB30_3_TXP
BL8	PCIE_4_TXN/USB30_3_TXN
BR4	PCIE_4_RXP/USB30_3_RXP
BR2	PCIE_4_RXN/USB30_3_RXN
BL13	PCIE_3_TXP
BL12	PCIE_3_TXN
BP3	PCIE_3_RXP
BP1	PCIE_3_RXN
BJ8	PCIE_2_TXP
BJ6	PCIE_2_TXN
BN4	PCIE_2_RXP
BN2	PCIE_2_RXN
BJ12	PCIE_1_TXP
BJ10	PCIE_1_TXN
BL2	PCIE_1_RXP
BL4	PCIE_1_RXN
BP34	RSVD_TP
BM34	RSVD_TP
BT36	RSVD_TP
BU15	RSVD_TP
BW15	RSVD_TP
AU33	RSVD_TP
BT34	RSVD_TP
CC38	RSVD
BY38	RSVD
BG33	PCH_IST_TP[1]
BK33	PCH_IST_TP[0]
BT50	PMC_PCH_PWROK

continued...

Ball#	External Name
BG1	PCH_OPIRCOMP
BC13	TP
BC15	TP
AM3	PCH_JTAG_TMS
AJ4	PCH_JTAG_TDO
AH3	PCH_JTAG_TDI
AL2	PCH_JTAG_TCK
CT2	RSVD
CP3	RSVD
AG4	PCH_JTAG_TRST_N
N1	CFG_RCOMP
N8	CFG_09
N9	CFG_08
W4	CFG_07
U2	CFG_06
U4	CFG_05
T1	CFG_04
R4	CFG_03
T3	CFG_02
N6	CFG_15
R8	CFG_14
U8	CFG_13
R12	CFG_12
R10	CFG_11
R6	CFG_10
P3	CFG_01
R2	CFG_00
U10	CFG_AVRB_STB_1P
P2	CFG_AVRB_STB_0P
U9	CFG_AVRB_STB_1N
P4	CFG_AVRB_STB_0N
AE4	BPM3_N
AF3	BPM2_N
AD1	BPM1_N
AE2	BPM0_N
AL4	PCH_JTAG_X

continued...

Ball#	External Name
BP51	INTRUDER_N
AU12	RSVD
AU13	RSVD
CH24	GP_H09/I2C4_SCL
CJ24	GP_H08/I2C4_SDA
CF24	GP_H07/I2C3_SCL
CC24	GP_H06/I2C3_SDA
CL25	GP_H05/I2C2_SCL
CM25	GP_H04/I2C2_SDA
CP23	GP_H03/SX_EXIT_HOLDOFF_N
CN23	GP_H02/MODEM_CLKREQ
CF21	GP_H19
CH21	GP_H18
CC21	GP_H17
CF23	GP_H16
CH23	GP_H15/AVS_I2S1_SCLK
CE23	GP_H14/AVS_I2S2_RXD
CC23	GP_H13/AVS_I2S2_TXD/ MODEM_CLKREQ
CB24	GP_H12/AVS_I2S2_SFRM/ CNV_RF_RESET_N
CB23	GP_H11/AVS_I2S2_SCLK
CL22	GP_H10/CPU_C10_GATE_N
CN22	GP_H01/SD_SDIO_PWR_EN_N/ CNV_RF_RESET_N
CN24	GP_H00
CG48	GP_G07/SD_SDIO_WP
CF49	GP_G06/SD_SDIO_CLK
CE50	GP_G05/SD_SDIO_CD_N
CG51	GP_G04/SD_SDIO_D3
CG49	GP_G03/SD_SDIO_D2
CJ50	GP_G02/SD_SDIO_D1
CJ47	GP_G01/SD_SDIO_D0
CG47	GP_G00/SD_SDIO_CMD
CP8	GP_E09/SML_CLK0/SATA_1_GP
CH3	GP_E08/SATA_0_GP
CH2	GP_E07/SATA_1_DEVSLP

continued...

Ball#	External Name
CH7	GP_E06/IMGCLKOUT_3
CJ3	GP_E05/SATA_LED_N
CM7	GP_E04/IMGCLKOUT_2
CL1	GP_E03/SATA_0_DEVSLP
CP7	GP_E23/CNV_RGI_RSP
CL4	GP_E22/CNV_RGI_DT
CJ1	GP_E21/CNV_BRI_RSP
CP4	GP_E20/CNV_BRI_DT
CK3	GP_E02/IMGCLKOUT_1
CM3	GP_E19/IMGCLKOUT_5/ PCIE_LNK_DOWN
CF5	GP_E18/DDI2_DDC_SDA
CF7	GP_E17/DDI2_DDC_SCL
CN4	GP_E16/DDI1_DDC_SDA
CK5	GP_E15/DDI1_DDC_SCL
CM5	GP_E14/DDI0_DDC_SDA
CL6	GP_E13/DDI0_DDC_SCL
CL8	GP_E12/IMGCLKOUT_4
CP6	GP_E11
CT7	GP_E10/SML_DATA0
CH9	GP_E01
CJ9	GP_E00/IMGCLKOUT_0
CM16	GP_D09/GSPI2_CLK/UART0A_TXD
CM14	GP_D08/GSPI2_SPI2_CS0_N/ UART0A_RXD
CR19	GP_D07
CL19	GP_D06
CE19	GP_D05
CC19	GP_D04
CE17	GP_D03/BK_3/SBK_3
CB21	GP_D23/I2C5_SCL
CB19	GP_D22/I2C5_SDA
CR22	GP_D21/CNV_PA_BLANKING
CM22	GP_D20/CNV_MFUART2_TXD
CF19	GP_D02/BK_2/SBK_2
CM20	GP_D19/CNV_MFUART2_RXD
CL14	GP_D18/AVS_I2S_MCLK

continued...

Ball#	External Name
CH19	GP_D17
CJ19	GP_D16
CP18	GP_D15/CNV_WCEN
CL17	GP_D14/GSPI2_CS1_N
CN19	GP_D13/I2C4B_SCL
CN18	GP_D12/I2C4B_SDA
CN17	GP_D11/GSPI2_MOSI/ UART0A_CTS_N
CR16	GP_D10/GSPI2_MISO/ UART0A_RTS_N
CJ17	GP_D01/BK_1/SBK_1
CH17	GP_D00/BK_0/SBK_0
CR11	GP_C09/UART0_TXD
CL11	GP_C08/UART0_RXD
CM9	GP_C07/PMC_SUSACK_N
CJ13	GP_C06/PMC_SUSWARN_N/ PMC_SUSPWRDNACK
CH13	GP_C05
CN11	GP_C04
CR14	GP_C03
CL12	GP_C23/UART2_CTS_N/ CNV_MFUART0_CTS_N
CE13	GP_C22/UART2_RTS_N/ CNV_MFUART0_RTS_N
CH15	GP_C21/UART2_TXD/ CNV_MFUART0_TXD
CF15	GP_C20/UART2_RXD/ CNV_MFUART0_RXD
CE15	GP_C02
CR9	GP_C19/I2C1_SCL
CP10	GP_C18/I2C1_SDA
CJ11	GP_C17/I2C0_SCL
CH11	GP_C16/I2C0_SDA
CF11	GP_C15/UART1_CTS_N
CC11	GP_C14/UART1_RTS_N
CE11	GP_C13/UART1_TXD
CC13	GP_C12/UART1_RXD
CT11	GP_C11/UART0_CTS_N
CM11	GP_C10/UART0_RTS_N
CN14	GP_C01
CN13	GP_C00
CN43	GP_B09/PCIE_CLKREQ4_N
CL41	GP_B08/PCIE_CLKREQ3_N
CR41	GP_B07/PCIE_CLKREQ2_N
CP46	GP_B06/PCIE_CLKREQ1_N
CN39	GP_B05/PCIE_CLKREQ0_N
CP48	GP_B04/CPU_GP_3
CR49	GP_B03/CPU_GP_2
CC36	GP_B23/DDI2_HPD/TIME_SYNC_0/ GSPI1_CS1_N
CF38	GP_B22/GSPI1_MOSI
CH38	GP_B21/GSPI1_MISO/NFC_CLKREQ
CE36	GP_B20/GSPI1_CLK/NFC_CLK
CR43	GP_B02/PMC_VRALERT_N
CH36	GP_B19/GSPI1_CS0_N
CL38	GP_B18/GSPI0_MOSI/UART2A_TXD
CP39	GP_B17/GSPI0_MISO/UART2A_RXD
CR38	GP_B16/GSPI0_CLK
CM38	GP_B15/GSPI0_CS0_N
CN38	GP_B14/SPKR_GSPI0_CS1_N
CM41	GP_B13/PMC_PLTRST_N
CN41	GP_B12/PMC_SLP_S0_N
CM45	GP_B11/PMCALERT_N
CT47	GP_B10/PCIE_CLKREQ5_N
CP45	GP_B01/PMC_CORE_VID1
CP44	GP_B00/PMC_CORE_VID0
CR44	GP_A09/SMB_ALERT_N
CN30	GP_A08/SMB_DATA
CR30	GP_A07/SMB_CLK
CN34	GP_A06/ESPI_RESET_N
CN35	GP_A05/ESPI_CLK
CR35	GP_A04/ESPI_CS_N
CM36	GP_A03/ESPI_IO_3
CL36	GP_A02/ESPI_IO_2
CM27	GP_A19/PCHHOT_N

continued...

Ball#	External Name
CR27	GP_A18/USB_OC0_N
CM30	GP_A17/DDI0_HPD
CN29	GP_A16/DDI1_HPD/TIME_SYNC_1
CP29	GP_A15
CN27	GP_A14/USB_OC3_N
CR33	GP_A13/USB_OC2_N
CL30	GP_A12/USB_OC1_N
CN33	GP_A11/CPU_GP_1
CM33	GP_A10/CPU_GP_0
CL33	GP_A01/ESPI_IO_1
CP34	GP_A00/ESPI_IO_0
CH28	GP_S07/DMIC_DATA_0
CF28	GP_S06/DMIC_CLK_0
CJ26	GP_S05/SNDW1_DATA
CE28	GP_S04/SNDW1_CLK
CF26	GP_S03/DMIC_DATA_1
CH26	GP_S02/DMIC_CLK_1
CE26	GP_S01
CC26	GP_S00
CB28	GP_R07/AVS_I2S1_TXD
CC30	GP_R06/AVS_I2S1_SFRM
CL28	GP_R05/HDA_SDI1/AVS_I2S1_RXD
CJ30	GP_R04/HDA_RST_N
CB30	GP_R03/HDA_SDI0/AVS_I2S0_RXD
CC28	GP_R02/HDA_SDO/AVS_I2S0_TXD
CF30	GP_R01/HDA_SYNC/ AVS_I2S0_SFRM
CH30	GP_R00/HDA_BCLK/AVS_I2S0_SCLK
CJ45	GP_F09/EMMC_DATA1
CJ41	GP_F08/EMMC_DATA0
CJ46	GP_F07/EMMC_CMD
CJ49	GP_F04/CNV_RF_RESET_N
CH40	GP_F18/EMMC_RESET_N
CF40	GP_F17/EMMC_CLK
CE38	GP_F16/EMMC_RCLK
CH42	GP_F15/EMMC_DATA7
	continued...

Ball#	External Name
CM46	GP_F14/EMMC_DATA6
CL44	GP_F13/EMMC_DATA5
CM43	GP_F12/EMMC_DATA4
CK45	GP_F11/EMMC_DATA3
CJ40	GP_F10/EMMC_DATA2
CA48	GP_DSW09/PMC_SLP_WLAN_N
BY49	GP_DSW08/PMC_SUSCLK
CB51	GP_DSW07
CB49	GP_DSW06/PMC_SLP_A_N
BW47	GP_DSW05/PMC_SLP_S4_N
CB47	GP_DSW04/PMC_SLP_S3_N
BU48	GP_DSW03/PMC_PWRBTN_N
CB48	GP_DSW02/LAN_WAKE_N
BW51	GP_DSW10/PMC_SLP_S5_N
CA50	GP_DSW01/PMC_ACPRESENT
BW48	GP_DSW00/PMC_BATLOW_N
CN52	RSVD
CT49	TP
CR52	TP
U15	RSVD
AK17	RSVD_TP
U33	RSVD_TP
W12	RSVD
W15	RSVD
AC17	RSVD_TP
AC18	RSVD_TP
AK33	RSVD_TP
U31	RSVD_TP
V31	RSVD_TP
AP33	RSVD_TP
AP34	RSVD_TP
V23	RSVD_TP
V25	RSVD_TP
AY18	IST_TRIG_1
AU17	IST_TRIG_0
AY17	IST_TP_1
	continued...

Ball#	External Name
AW17	IST_TP_0
AR1	RSVD_TP
CN50	EMMC_RCOMP
BT38	RSVD_TP
BP49	PMC_DSW_PWROK
W2	MDSI_DE_TE_2
BV41	PMC_DRAM_RESET_N
E3	DISP_RCOMP
E1	RSVD_TP
AA4	DISP_UTILS_MDSI_DE_TE_1
BK48	LP4x_3_DQS3_DP/DDR_1_DQS7_DP
BE48	LP4x_3_DQS1_DP/DDR_1_DQS5_DP
AJ48	LP4x_2_DQS3_DP/DDR_1_DQS3_DP
AC48	LP4x_2_DQS1_DP/DDR_1_DQS1_DP
V48	LP4x_1_DQS3_DP/DDR_0_DQS7_DP
N48	LP4x_1_DQS1_DP/DDR_0_DQS5_DP
C32	LP4x_0_DQS3_DP/DDR_0_DQS3_DP
C26	LP4x_0_DQS1_DP/DDR_0_DQS1_DP
BP42	LP4x_3_DQS2_DP/DDR_1_DQS6_DP
BF42	LP4x_3_DQS0_DP/DDR_1_DQS4_DP
AH42	LP4x_2_DQS2_DP/DDR_1_DQS2_DP
AD42	LP4x_2_DQS0_DP/DDR_1_DQS0_DP
T42	LP4x_1_DQS2_DP/DDR_0_DQS6_DP
K42	LP4x_1_DQS0_DP/DDR_0_DQS4_DP
L36	LP4x_0_DQS2_DP/DDR_0_DQS2_DP
L30	LP4x_0_DQS0_DP/DDR_0_DQS0_DP
BK50	LP4x_3_DQS3_DN/ DDR_1_DQS7_DN
BE50	LP4x_3_DQS1_DN/ DDR_1_DQS5_DN
AJ50	LP4x_2_DQS3_DN/ DDR_1_DQS3_DN
AC50	LP4x_2_DQS1_DN/ DDR_1_DQS1_DN
V50	LP4x_1_DQS3_DN/ DDR_0_DQS7_DN
M50	LP4x_1_DQS1_DN/ DDR_0_DQS5_DN

continued...

Ball#	External Name
D32	LP4x_0_DQS3_DN/ DDR_0_DQS3_DN
D26	LP4x_0_DQS1_DN/ DDR_0_DQS1_DN
BM42	LP4x_3_DQS2_DN/ DDR_1_DQS6_DN
BH42	LP4x_3_DQS0_DN/ DDR_1_DQS4_DN
AK42	LP4x_2_DQS2_DN/ DDR_1_DQS2_DN
AB42	LP4x_2_DQS0_DN/ DDR_1_DQS0_DN
V42	LP4x_1_DQS2_DN/ DDR_0_DQS6_DN
M42	LP4x_1_DQS0_DN/ DDR_0_DQS4_DN
L34	LP4x_0_DQS2_DN/ DDR_0_DQS2_DN
L28	LP4x_0_DQS0_DN/ DDR_0_DQS0_DN
BJ47	LP4x_3_DQ31/DDR_1_DQ63
BH48	LP4x_3_DQ30/DDR_1_DQ62
BJ49	LP4x_3_DQ29/DDR_1_DQ61
BM51	LP4x_3_DQ28/DDR_1_DQ60
BJ51	LP4x_3_DQ27/DDR_1_DQ59
BL47	LP4x_3_DQ26/DDR_1_DQ58
BL49	LP4x_3_DQ25/DDR_1_DQ57
BM48	LP4x_3_DQ24/DDR_1_DQ56
BC47	LP4x_3_DQ15/DDR_1_DQ47
BC48	LP4x_3_DQ14/DDR_1_DQ46
BD49	LP4x_3_DQ13/DDR_1_DQ45
BF51	LP4x_3_DQ12/DDR_1_DQ44
BC51	LP4x_3_DQ11/DDR_1_DQ43
BF47	LP4x_3_DQ10/DDR_1_DQ42
BF49	LP4x_3_DQ09/DDR_1_DQ41
BF48	LP4x_3_DQ08/DDR_1_DQ40
AG47	LP4x_2_DQ31/DDR_1_DQ31
AG48	LP4x_2_DQ30/DDR_1_DQ30
AH49	LP4x_2_DQ29/DDR_1_DQ29
AK47	LP4x_2_DQ28/DDR_1_DQ28
AG51	LP4x_2_DQ27/DDR_1_DQ27

continued...

Ball#	External Name
AK51	LP4x_2_DQ26/DDR_1_DQ26
AK48	LP4x_2_DQ25/DDR_1_DQ25
AK49	LP4x_2_DQ24/DDR_1_DQ24
AB47	LP4x_2_DQ15/DDR_1_DQ15
AB48	LP4x_2_DQ14/DDR_1_DQ14
AB49	LP4x_2_DQ13/DDR_1_DQ13
AD47	LP4x_2_DQ12/DDR_1_DQ12
AB51	LP4x_2_DQ11/DDR_1_DQ11
AE51	LP4x_2_DQ10/DDR_1_DQ10
AE48	LP4x_2_DQ09/DDR_1_DQ09
AE49	LP4x_2_DQ08/DDR_1_DQ08
T47	LP4x_1_DQ31/DDR_0_DQ63
T48	LP4x_1_DQ30/DDR_0_DQ62
U49	LP4x_1_DQ29/DDR_0_DQ61
W47	LP4x_1_DQ28/DDR_0_DQ60
U51	LP4x_1_DQ27/DDR_0_DQ59
W51	LP4x_1_DQ26/DDR_0_DQ58
Y48	LP4x_1_DQ25/DDR_0_DQ57
W49	LP4x_1_DQ24/DDR_0_DQ56
K50	LP4x_1_DQ15/DDR_0_DQ47
L48	LP4x_1_DQ14/DDR_0_DQ46
K49	LP4x_1_DQ13/DDR_0_DQ45
N47	LP4x_1_DQ12/DDR_0_DQ44
L51	LP4x_1_DQ11/DDR_0_DQ43
P51	LP4x_1_DQ10/DDR_0_DQ42
P48	LP4x_1_DQ09/DDR_0_DQ41
P49	LP4x_1_DQ08/DDR_0_DQ40
F30	LP4x_0_DQ31/DDR_0_DQ31
E30	LP4x_0_DQ30/DDR_0_DQ30
D31	LP4x_0_DQ29/DDR_0_DQ29
B33	LP4x_0_DQ28/DDR_0_DQ28
B30	LP4x_0_DQ27/DDR_0_DQ27
F33	LP4x_0_DQ26/DDR_0_DQ26
D33	LP4x_0_DQ25/DDR_0_DQ25
E33	LP4x_0_DQ24/DDR_0_DQ24
F25	LP4x_0_DQ15/DDR_0_DQ15
E25	LP4x_0_DQ14/DDR_0_DQ14
D25	LP4x_0_DQ13/DDR_0_DQ13
B28	LP4x_0_DQ12/DDR_0_DQ12
B25	LP4x_0_DQ11/DDR_0_DQ11
F27	LP4x_0_DQ10/DDR_0_DQ10
D28	LP4x_0_DQ09/DDR_0_DQ09
E28	LP4x_0_DQ08/DDR_0_DQ08
BP39	LP4x_3_DQ23/DDR_1_DQ55
BP41	LP4x_3_DQ22/DDR_1_DQ54
BP44	LP4x_3_DQ21/DDR_1_DQ53
BM45	LP4x_3_DQ20/DDR_1_DQ52
BP45	LP4x_3_DQ19/DDR_1_DQ51
BM39	LP4x_3_DQ18/DDR_1_DQ50
BM44	LP4x_3_DQ17/DDR_1_DQ49
BM41	LP4x_3_DQ16/DDR_1_DQ48
BH39	LP4x_3_DQ07/DDR_1_DQ39
BH41	LP4x_3_DQ06/DDR_1_DQ38
BH44	LP4x_3_DQ05/DDR_1_DQ37
BF45	LP4x_3_DQ04/DDR_1_DQ36
BH45	LP4x_3_DQ03/DDR_1_DQ35
BF39	LP4x_3_DQ02/DDR_1_DQ34
BF44	LP4x_3_DQ01/DDR_1_DQ33
BF41	LP4x_3_DQ00/DDR_1_DQ32
AK41	LP4x_2_DQ23/DDR_1_DQ23
AK39	LP4x_2_DQ22/DDR_1_DQ22
AK44	LP4x_2_DQ21/DDR_1_DQ21
AH44	LP4x_2_DQ20/DDR_1_DQ20
AK45	LP4x_2_DQ19/DDR_1_DQ19
AH41	LP4x_2_DQ18/DDR_1_DQ18
AH45	LP4x_2_DQ17/DDR_1_DQ17
AH39	LP4x_2_DQ16/DDR_1_DQ16
AD39	LP4x_2_DQ07/DDR_1_DQ07
AD41	LP4x_2_DQ06/DDR_1_DQ06
AD44	LP4x_2_DQ05/DDR_1_DQ05
AB45	LP4x_2_DQ04/DDR_1_DQ04
AD45	LP4x_2_DQ03/DDR_1_DQ03

continued...

Ball#	External Name
AB44	LP4x_2_DQ02/DDR_1_DQ02
AB39	LP4x_2_DQ01/DDR_1_DQ01
AB41	LP4x_2_DQ00/DDR_1_DQ00
V39	LP4x_1_DQ23/DDR_0_DQ55
V41	LP4x_1_DQ22/DDR_0_DQ54
V44	LP4x_1_DQ21/DDR_0_DQ53
T45	LP4x_1_DQ20/DDR_0_DQ52
V45	LP4x_1_DQ19/DDR_0_DQ51
T39	LP4x_1_DQ18/DDR_0_DQ50
T44	LP4x_1_DQ17/DDR_0_DQ49
T41	LP4x_1_DQ16/DDR_0_DQ48
M45	LP4x_1_DQ07/DDR_0_DQ39
M44	LP4x_1_DQ06/DDR_0_DQ38
M41	LP4x_1_DQ05/DDR_0_DQ37
K45	LP4x_1_DQ04/DDR_0_DQ36
M39	LP4x_1_DQ03/DDR_0_DQ35
K39	LP4x_1_DQ02/DDR_0_DQ34
K44	LP4x_1_DQ01/DDR_0_DQ33
K41	LP4x_1_DQ00/DDR_0_DQ32
M36	LP4x_0_DQ23/DDR_0_DQ23
P36	LP4x_0_DQ22/DDR_0_DQ22
J36	LP4x_0_DQ21/DDR_0_DQ21
H34	LP4x_0_DQ20/DDR_0_DQ20
H36	LP4x_0_DQ19/DDR_0_DQ19
P34	LP4x_0_DQ18/DDR_0_DQ18
J34	LP4x_0_DQ17/DDR_0_DQ17
M34	LP4x_0_DQ16/DDR_0_DQ16
P30	LP4x_0_DQ07/DDR_0_DQ07
M30	LP4x_0_DQ06/DDR_0_DQ06
J30	LP4x_0_DQ05/DDR_0_DQ05
H28	LP4x_0_DQ04/DDR_0_DQ04
H30	LP4x_0_DQ03/DDR_0_DQ03
P28	LP4x_0_DQ02/DDR_0_DQ02
J28	LP4x_0_DQ01/DDR_0_DQ01
M28	LP4x_0_DQ00/DDR_0_DQ00
BB45	DDR_1_PAR

continued...

Ball#	External Name
AM48	DDR_1_ODT1
BA48	DDR_1_ODT0
AT42	LP4x_2_CA1/DDR_1_MA09
AT41	LP4x_2_CA3/DDR_1_MA08
AT39	LP4x_2_CA4/DDR_1_MA07
AT44	LP4x_2_CA2/DDR_1_MA06
AP41	LP4x_2_CA0/DDR_1_MA05
AY44	DDR_1_MA04
BB42	DDR_1_MA03
AR47	LP4x_3_CA5/DDR_1_MA02
AR49	LP4x_3_CA3/DDR_1_MA16_RAS_N
BA49	LP4x_3_CA1/DDR_1_MA15_CAS_N
AP50	LP4x_3_CA2/DDR_1_MA14_WE_N
AT51	LP4x_3_CA0/DDR_1_MA13
AP45	DDR_1_MA12
AP42	DDR_1_MA11
AT48	DDR_1_MA10
AN49	DDR_1_MA01
AP48	DDR_1_MA00
AY47	LP4x_3_CS0
AY50	LP4x_2_CS1
BA51	LP4x_3_CS1/DDR_1_CS1_N
AW48	LP4x_2_CS0/DDR_1_CS0_N
AV49	LP4x_3_CLK_DP/DDR_1_CLK1_DP
AV48	LP4x_2_CLK_DP/DDR_1_CLK0_DP
AV51	LP4x_3_CLK_DN/DDR_1_CLK1_DN
AV47	LP4x_2_CLK_DN/DDR_1_CLK0_DN
BB44	LP4x_3_CKE0
AY41	LP4x_2_CKE1
AY45	LP4x_3_CKE1/DDR_1_CKE1
AY42	LP4x_2_CKE0/DDR_1_CKE0
AP44	DDR_1_BG1
AT45	LP4x_2_CA5/DDR_1_BG0
AN47	DDR_1_BA1
AN51	LP4x_3_CA4/DDR_1_BA0
AY39	DDR_1_ALERT_N

continued...

Ball#	External Name
AP39	DDR_1_ACT_N
BV45	DDR0_VREF_DQ
G42	DDR_0_PAR
D47	DDR_0_ODT1
J49	DDR_0_ODT0
E39	LP4x_0_CA1/DDR_0_MA09
D38	LP4x_0_CA3/DDR_0_MA08
B39	LP4x_0_CA4/DDR_0_MA07
D37	LP4x_0_CA2/DDR_0_MA06
E35	LP4x_0_CA0/DDR_0_MA05
C43	DDR_0_MA04
E41	DDR_0_MA03
C44	LP4x_1_CA5/DDR_0_MA02
A45	LP4x_1_CA3/DDR_0_MA16_RAS_N
F49	LP4x_1_CA1/DDR_0_MA15_CAS_N
C46	LP4x_1_CA2/DDR_0_MA14_WE_N
J48	LP4x_1_CA0/DDR_0_MA13
B36	DDR_0_MA12
C37	DDR_0_MA11
A47	DDR_0_MA10
E43	DDR_0_MA01
C45	DDR_0_MA00
G50	LP4x_1_CS0
H50	LP4x_0_CS1
F52	LP4x_1_CS1/DDR_0_CS1_N
D45	LP4x_0_CS0/DDR_0_CS0_N
E48	LP4x_1_CLK_DP/DDR_0_CLK1_DP
E46	LP4x_0_CLK_DP/DDR_0_CLK0_DP
D49	LP4x_1_CLK_DN/DDR_0_CLK1_DN
F45	LP4x_0_CLK_DN/DDR_0_CLK0_DN
G48	LP4x_1_CKE0
B41	LP4x_0_CKE1
D42	LP4x_1_CKE1/DDR_0_CKE1
F41	LP4x_0_CKE0/DDR_0_CKE0
D36	DDR_0_BG1
F38	LP4x_0_CA5/DDR_0_BG0
E50	DDR_0_BA1
B44	LP4x_1_CA4/DDR_0_BA0
D41	DDR_0_ALERT_N
F36	DDR_0_ACT_N
BB41	DDR_VTT_CTL
BV42	DDR_1_VREF_CA
BV44	DDR_0_VREF_CA
A51	RSVD_TP
B52	RSVD_TP
C48	DDR_RCOMP2
D50	DDR_RCOMP1
C49	DDR_RCOMP0
C8	DDI2_TXP3
B7	DDI2_TXP2
C10	DDI2_TXP1
E12	DDI2_TXP0
A8	DDI2_TXN3
A7	DDI2_TXN2
D10	DDI2_TXN1
D11	DDI2_TXN0
B9	DDI2_AUXP
D8	DDI2_AUXN
F3	DDI1_TXP3
J4	DDI1_TXP2
H3	DDI1_TXP1
L4	DDI1_TXP0
G1	DDI1_TXN3
J2	DDI1_TXN2
H4	DDI1_TXN1
M2	DDI1_TXN0
K1	DDI1_AUXP
K3	DDI1_AUXN
J8	DDI0_TXP3
H6	DDI0_TXP2
E5	DDI0_TXP1
F6	DDI0_TXP0

continued...

Ball#	External Name
J7	DDI0_TXN3
J5	DDI0_TXN2
D4	DDI0_TXN1
G7	DDI0_TXN0
F9	DDI0_AUXP
E10	DDI0_AUXN
AN3	DBG_PMODE
AT2	MCSI_RCOMP
AM12	MCSI_D_D1P_C_D2P
AM10	MCSI_D_D1N_C_D2N
AR8	MCSI_D_D0P_C_D3P
AR6	MCSI_D_D0N_C_D3N
AR12	MCSI_D_CKP
AR10	MCSI_D_CKN
AG12	MCSI_C_D0P
AG10	MCSI_C_D0N
AJ9	MCSI_C_D1P
AJ8	MCSI_C_D1N
AM9	MCSI_C_CKP
AM8	MCSI_C_CKN
AD9	MCSI_B_D0P_A_D3P
AD8	MCSI_B_D0N_A_D3N
AD13	MCSI_B_D1P_A_D2P
AD12	MCSI_B_D1N_A_D2N
AG8	MCSI_B_CKP
AG6	MCSI_B_CKN
AB9	MCSI_A_D1P
AB8	MCSI_A_D1N
W8	MCSI_A_D0P
W6	MCSI_A_D0N
AB13	MCSI_A_CKP
AB12	MCSI_A_CKN
AM2	PROC_PWR_GD
CK50	PROC_POPIRCOMP
BD33	RSVD_TP
BE33	RSVD_TP
AG2	CPU_JTAG_TRST_N
AM1	CPU_JTAG_TMS
AJ2	CPU_JTAG_TDO
AH1	CPU_JTAG_TDI
AK3	CPU_JTAG_TCK
D3	RSVD
B3	RSVD
AR17	RSVD_TP
CE40	RSVD_TP
CL52	CNV_WT_RCOMP
BY41	CNV_WT_D1P
BY39	CNV_WT_D1N
CC41	CNV_WT_D0P
CC42	CNV_WT_D0N
BY45	CNV_WT_CLKP
BY44	CNV_WT_CLKN
CG44	CNV_WR_D1P
CG46	CNV_WR_D1N
CE44	CNV_WR_D0P
CE45	CNV_WR_D0N
CD42	CNV_WR_CLKP
CD41	CNV_WR_CLKN
AW10	CLKOUT_PCIE_P4
AW6	CLKOUT_PCIE_P3
BA9	CLKOUT_PCIE_P2
BA13	CLKOUT_PCIE_P1
BC6	CLKOUT_PCIE_P0
AW12	CLKOUT_PCIE_N4
AW8	CLKOUT_PCIE_N3
BA8	CLKOUT_PCIE_N2
BA12	CLKOUT_PCIE_N1
BC8	CLKOUT_PCIE_N0
BG3	ICLK_BIASREF
BC10	RSVD_TP
BC12	RSVD_TP

continued...

Ball#	External Name
AB3	CATERR_N
CB17	eDP_BKLTEN
CC15	eDP_BKLTCTL