



Intel[®] 300 Series and Intel[®] C240 Series Chipset Family Platform Controller Hub

Datasheet – Volume 1 of 2

Revision 009

September 2021



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.

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

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

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

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

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

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

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

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

© Intel Corporation. All Rights Reserved.



Contents

1	Introduction	28
1.1	About this Manual	28
1.2	Overview	28
1.3	PCH SKUs	29
2	PCH Controller Device IDs	36
2.1	Device and Revision ID Table	36
3	Flexible I/O	38
3.1	Overview	38
3.2	Flexible I/O Implementation	38
3.2.1	PCH-H	38
3.3	Flexible I/O Lane Selection	39
3.3.1	PCIe*/SATA Lane Selection.....	39
4	Memory Mapping	40
4.1	Overview	40
4.2	Functional Description	40
4.2.1	PCI Devices and Functions	40
4.2.2	Fixed I/O Address Ranges	42
4.2.3	Variable I/O Decode Ranges	44
4.3	Memory Map.....	45
4.3.1	Boot Block Update Scheme	47
5	System Management	49
5.1	Acronyms.....	49
5.2	Overview	49
5.3	Features	49
5.3.1	Theory of Operation.....	49
5.3.2	TCO Modes.....	50
6	High Precision Event Timer (HPET)	52
6.1	Overview	52
6.1.1	Timer Accuracy	52
6.1.2	Timer Off-load	52
6.1.3	Interrupt Mapping	54
6.1.4	Periodic Versus Non-Periodic Modes	55
6.1.5	Enabling the Timers.....	56
6.1.6	Interrupt Levels	57
6.1.7	Handling Interrupts	57
6.1.8	Issues Related to 64-Bit Timers with 32-Bit Processors.....	57
7	Thermal Management	58
7.1	PCH Thermal Sensor	58
7.1.1	Modes of Operation	58
7.1.2	Temperature Trip Point	58
7.1.3	Thermal Sensor Accuracy (Taccuracy)	58
7.1.4	Thermal Reporting to an EC.....	58
7.1.5	Thermal Trip Signal (PCHHOT#)	59
8	Power and Ground Signals	60
9	Pin Straps	62



10	Electrical and Thermal Characteristics	66
10.1	Power Rail Absolute Maximum Ratings	66
10.2	Thermal Specification	66
10.3	General DC Characteristics	66
10.4	AC Characteristics	82
10.4.1	Panel Power Sequencing and Backlight Control	85
10.5	Overshoot/Undershoot Guidelines	105
11	Ballout Definition	107
12	8254 Timers	120
12.1	Overview	120
12.1.1	Timer Programming	120
12.1.2	Reading from Interval Timer	121
13	Audio, Voice, and Speech	123
13.1	Acronyms	123
13.2	AVS Subsystem Overview	123
13.3	Signal Description	124
13.4	Integrated Pull-Ups and Pull-Downs	126
13.5	I/O Signal Planes and States	126
13.6	AVS Feature Summary	127
13.6.1	Intel High Definition Audio Controller Capabilities	127
13.6.2	Audio DSP Capabilities	128
13.6.3	Intel High Definition Audio Link Capabilities	128
13.6.4	Intel Display Audio Link Capabilities	128
13.6.5	DMIC Interface	128
13.6.6	I ² S/PCM Interface	129
13.6.7	SoundWire Interface	129
14	Controller Link	130
14.1	Overview	130
14.2	Signal Description	130
14.3	Integrated Pull-Ups and Pull-Downs	130
14.4	I/O Signal Planes and States	130
14.5	External CL_RST# Pin Driven/Open-drain Mode Support	131
15	Processor Sideband Signals	132
15.1	Acronyms	132
15.2	Overview	132
15.3	Signal Description	132
15.4	Integrated Pull-Ups and Pull-Downs	132
15.5	I/O Signal Planes and States	132
15.6	Functional Description	133
16	Digital Display Signals	134
16.1	Acronyms	134
16.2	Signal Description	134
16.3	Embedded DisplayPort* (eDP*) Backlight Control Signals	135
16.4	Integrated Pull-Ups and Pull-Downs	135
16.5	I/O Signal Planes and States	135
17	Enhanced Serial Peripheral Interface (eSPI)	137
17.1	Acronyms	137
17.2	Overview	137
17.3	Signal Description	137
17.4	Integrated Pull-Ups and Pull-Downs	138
17.5	I/O Signal Planes and States	138



17.6	Functional Description	138
17.6.1	Features	138
17.6.2	Protocols	138
17.6.3	WAIT States from eSPI Slave	139
17.6.4	In-Band Link Reset	139
17.6.5	Slave Discovery	139
17.6.6	Flash Sharing Mode	140
17.6.7	Channels and Supported Transactions	140
18	General Purpose Input and Output (GPIO)	146
18.1	Acronyms	146
18.2	Overview	146
18.3	Signal Description	147
18.4	Functional Description	155
18.4.1	GPIO Voltage	155
18.4.2	Programmable Hardware Debouncer	156
18.4.3	Integrated Pull-ups and Pull-downs	156
18.4.4	GPPJ_RCOMP_1P8 Signal	156
18.4.5	Interrupt / IRQ via GPIO Requirement	156
18.4.6	SMI#/SCI and NMI	156
18.4.7	Timed GPIO	156
18.4.8	GPIO Blink (BK) and Serial Blink (SBK)	157
18.4.9	GPIO Ownership	157
19	Intel® Serial I/O Generic SPI (GSPI) Controllers	158
19.1	Acronyms	158
19.2	Overview	158
19.3	Signal Description	158
19.4	Integrated Pull-Ups and Pull-Downs	159
19.5	I/O Signal Planes and States	159
19.6	Functional Description	159
19.6.1	Features	159
19.6.2	Controller Overview	160
19.6.3	DMA Controller	160
19.6.4	Reset	161
19.6.5	Power Management	161
19.6.6	Interrupts	162
19.6.7	Error Handling	162
20	Intel® Serial I/O Inter-Integrated Circuit (I²C) Controllers	163
20.1	Acronyms	163
20.2	References	163
20.3	Overview	163
20.4	Signal Description	163
20.5	I/O Signal Planes and States	164
20.6	Functional Description	164
20.6.1	Features	164
20.6.2	Protocols Overview	164
20.6.3	DMA Controller	165
20.6.4	Reset	166
20.6.5	Power Management	166
20.6.6	Interrupts	167
20.6.7	Error Handling	167
20.6.8	Programmable SDA Hold Time	167
21	Gigabit Ethernet Controller	168



21.1	Acronyms	168
21.2	References.....	168
21.3	Overview	168
21.4	Signal Description	168
21.5	Integrated Pull-Ups and Pull-Downs.....	169
21.6	I/O Signal Planes and States	169
21.7	Functional Description	170
21.7.1	GbE PCI Express* Bus Interface.....	171
21.7.2	Error Events and Error Reporting	172
21.7.3	Ethernet Interface	172
21.7.4	PCI Power Management	173
22	Interrupt Interface	174
22.1	Acronyms	174
22.2	Overview	174
22.3	Signal Description.....	174
22.4	Integrated Pull-Ups and Pull-Downs.....	174
22.5	I/O Signal Planes and States	174
22.6	Functional Description	175
22.6.1	8259 Interrupt Controllers (PIC)	178
22.6.2	Interrupt Handling	179
22.6.3	Initialization Command Words (ICWx)	180
22.6.4	Operation Command Words (OCW)	181
22.6.5	Modes of Operation	181
22.6.6	Masking Interrupts.....	183
22.6.7	Steering PCI Interrupts.....	184
22.7	Advanced Programmable Interrupt Controller (APIC) (D31:F0).....	184
22.7.1	Interrupt Handling	184
22.7.2	Interrupt Mapping.....	184
22.7.3	PCI/PCI Express* Message-Based Interrupts	184
22.7.4	IOxAPIC Address Remapping.....	185
22.7.5	External Interrupt Controller Support	185
22.8	Serial Interrupt	185
22.8.1	Start Frame	186
22.8.2	Stop Frame.....	186
22.8.3	Specific Interrupts Not Supported Using SERIRQ	187
23	Integrated Sensor Hub (ISH)	188
23.1	Acronyms	188
23.2	References.....	188
23.3	Overview	188
23.4	Signal Description	189
23.5	Integrated Pull-Ups and Pull-Downs.....	189
23.6	I/O Signal Planes and States	190
23.7	Functional Description	190
23.7.1	ISH Micro-Controller.....	190
23.7.2	SRAM.....	190
23.7.3	PCI Host Interface	191
23.7.4	Power Domains and Management.....	191
23.7.5	ISH IPC	191
23.7.6	ISH Interrupt Handling via IOAPIC (Interrupt Controller).....	192
23.7.7	ISH I2C Controllers	192
23.7.8	ISH UART Controller.....	192
23.7.9	ISH GSPI Controller	193
23.7.10	ISH GPIOs	193



24	Low Pin Count (LPC)	194
24.1	Acronyms.....	194
24.2	References	194
24.3	Overview	194
24.4	Signal Description	194
24.5	Integrated Pull-Ups and Pull-Downs	195
24.6	I/O Signal Planes and States.....	195
24.7	Functional Description.....	195
24.7.1	LPC Cycle Types.....	195
24.7.2	Start Field Definition	196
24.7.3	Cycle Type/Direction (CYCTYPE + DIR).....	196
24.7.4	Size.....	197
24.7.5	SYNC Timeout.....	197
24.7.6	SYNC Error Indication	197
24.7.7	LFRAME# Usage.....	198
24.7.8	I/O Cycles.....	198
24.7.9	LPC Power Management	198
24.7.10	Configuration and PCH Implications	198
25	PCH and System Clocks	199
25.1	Overview	199
25.2	PCH ICC Clocking Profiles	199
25.3	PCH ICC XTAL Input Configurations	200
25.4	Signal Descriptions.....	200
25.5	I/O Signal Pin States	201
25.6	General Features.....	202
26	PCI Express* (PCIe*)	203
26.1	Overview	203
26.2	Signal Description	204
26.3	I/O Signal Planes and States.....	204
26.4	PCI Express* Port Support Feature Details.....	204
26.4.1	Intel® Rapid Storage Technology (Intel® RST) for PCIe* Storage.....	206
26.4.2	Interrupt Generation.....	206
26.4.3	PCI Express* Power Management	207
26.4.4	Dynamic Link Throttling	208
26.4.5	Port 8xh Decode	209
26.4.6	Separate Reference Clock with Independent SSC (SRIS).....	209
26.4.7	Advanced Error Reporting.....	210
26.4.8	Single- Root I/O Virtualization (SR- IOV)	210
26.4.9	SERR# Generation	210
26.4.10	Hot-Plug	210
26.4.11	PCI Express* Lane Polarity Inversion	211
26.4.12	PCI Express* Controller Lane Reversal	211
26.4.13	Precision Time Measurement (PTM).....	212
27	Power Management	213
27.1	Acronyms.....	213
27.2	References	213
27.3	Overview	213
27.4	Signal Description	213
27.5	Integrated Pull-Ups and Pull-Downs	216
27.6	I/O Signal Planes and States.....	216
27.7	Functional Description.....	218
27.7.1	Features	218
27.7.2	PCH and System Power States	218



27.7.3	System Power Planes	220
27.7.4	SMI#/SCI Generation	220
27.7.5	C-States	223
27.7.6	Dynamic 24 MHz Clock Control	223
27.7.7	Sleep States	224
27.7.8	Event Input Signals and Their Usage	228
27.7.9	ALT Access Mode	232
27.7.10	System Power Supplies, Planes, and Signals	233
27.7.11	Legacy Power Management Theory of Operation	238
27.7.12	Reset Behavior	238
28	Real Time Clock (RTC)	241
28.1	Acronyms	241
28.2	Overview	241
28.3	Signal Description	241
28.4	I/O Signal Planes and States	242
28.5	Functional Description	242
28.5.1	Update Cycles	243
28.5.2	Interrupts	243
28.5.3	Lockable RAM Ranges	243
28.5.4	Century Rollover	243
28.5.5	Clearing Battery-Backed RTC RAM	244
28.5.6	External RTC Circuitry	244
29	Serial ATA (SATA)	245
29.1	Acronyms	245
29.2	References	245
29.3	Overview	245
29.4	Signals Description	245
29.5	Integrated Pull-Ups and Pull-Downs	249
29.6	I/O Signal Planes and States	249
29.7	Functional Description	250
29.7.1	SATA 6 Gb/s Support	250
29.7.2	SATA Feature Support	250
29.7.3	Hot-Plug Operation	250
29.7.4	Intel® Rapid Storage Technology (Intel® RST)	250
29.7.5	Intel® Rapid Storage Technology enterprise (Intel® RSTe) - for Server/Workstation Only	252
29.7.6	Power Management Operation	253
29.7.7	SATA Device Presence	255
29.7.8	SATA LED	256
29.7.9	Advanced Host Controller Interface (AHCI) Operation	256
29.7.10	Enclosure Management (SGPIO Signals)	257
30	System Management Interface and SMLink	260
30.1	Acronyms	260
30.2	References	260
30.3	Overview	260
30.4	Signal Description	260
30.5	Integrated Pull-Ups and Pull-Downs	261
30.6	I/O Signal Planes and States	261
30.7	Functional Description	261
31	Host System Management Bus (SMBus) Controller	262
31.1	Acronyms	262
31.2	References	262
31.3	Overview	262



31.4	Signal Description	262
31.5	Integrated Pull-Ups and Pull-Downs	262
31.6	I/O Signal Planes and States	263
31.7	Functional Description	263
31.7.1	Host Controller	263
31.7.2	SMBus Slave Interface	269
31.8	SMBus Power Gating	276
32	Serial Peripheral Interface (SPI)	277
32.1	Acronyms	277
32.2	Overview	277
32.3	Signal Description	277
32.4	Integrated Pull-Ups and Pull-Downs	278
32.5	I/O Signal Planes and States	278
32.6	Functional Description	279
32.6.1	SPI0 for Flash	279
32.6.2	SPI0 Support for TPM	283
32.6.3	SPI1 Support for Touch Device	283
33	Testability	284
33.1	JTAG	284
33.1.1	Acronyms	284
33.1.2	References	284
33.1.3	Overview	284
33.1.4	Signal Description	284
33.1.5	I/O Signal Planes and States	285
33.2	Intel® Trace Hub (Intel® TH)	285
33.2.1	Overview	285
33.2.2	Platform Setup	286
33.3	Direct Connect Interface (DCI)	286
33.3.1	Out Of Band (OOB) Hosting DCI	287
33.3.2	USB Hosting DCI.DBC	287
33.3.3	Platform Setup	287
34	Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers... 288	
34.1	Acronyms	288
34.2	Overview	288
34.3	Signal Description	288
34.4	Integrated Pull-Ups and Pull-Downs	289
34.5	I/O Signal Planes and States	289
34.6	Functional Description	289
34.6.1	Features	289
34.6.2	UART Serial (RS-232) Protocols Overview	290
34.6.3	16550 8-bit Addressing - Debug Driver Compatibility	291
34.6.4	DMA Controller	291
34.6.5	Reset	292
34.6.6	Power Management	292
34.6.7	Interrupts	293
34.6.8	Error Handling	293
35	Universal Serial Bus (USB)	294
35.1	Acronyms	294
35.2	References	294
35.3	Overview	294
35.4	Signal Description	294
35.5	Integrated Pull-Ups and Pull-Downs	297



35.6	I/O Signal Planes and States	297
35.7	Functional Description	298
35.7.1	eXtensible Host Controller Interface (xHCI) Controller (D20:F0)	298
35.7.2	USB Dual Role Support - eXtensible Device Controller Interface (xHCI) Controller (D20:F1)	298
35.7.3	Supported USB 2.0 Ports	298
36	Connectivity Integrated (CNVi)	299
36.1	Acronyms	299
36.2	References	299
36.3	Overview	299
36.4	Signal Description	299
36.5	I/O Signal Planes and States	301
36.6	Functional Description	303
37	GPIO Serial Expander	304
37.1	Acronyms, Definitions	304
37.2	Overview	304
37.3	Signal Description	304
37.4	Integrated Pull-ups and Pull-downs	304
37.5	Functional Description	304
38	Direct Media Interface	306
38.1	Acronyms	306
38.2	Overview	306
38.3	Signal Description	306
38.4	I/O Signal Planes and States	306
38.5	Functional Description	306
38.5.1	Lane Reversal	306
38.5.2	Polarity Inversion	306
39	Private Configuration Space Target Port ID	307
39.1	Overview	307
40	Secure Digital eXtended Capacity (SDXC)	309
40.1	Acronyms	309
40.2	References	309
40.3	Overview	309
40.4	Signal Description	309
40.5	I/O Signal Planes and States	310
40.6	Functional Description	310
40.7	Virtual GPIO (vGPIO) Used as Interrupt	310

Figures

3-1	High Speed I/O (HSIO) Lane Multiplexing in PCH-H	36
5-1	TCO Compatible Mode SMBus Configuration	48
5-2	Advanced TCO Mode	49
10-1	PCI Express* Transmitter Eye	82
10-2	PCI Express* Receiver Eye	82
10-3	Panel Power Sequencing	83
10-4	Clock Timing	87
10-5	Measurement Points for Differential Waveforms	88
10-6	I2C, SMBus/SMLink Transaction	89
10-7	USB Rise and Fall Times	90
10-8	USB Jitter	91
10-9	USB EOP Width	91



10-10	SMBus/SMLink Timeout	93
10-11	Intel® High Definition Audio (Intel® HD Audio) Input and Output Timings.....	94
10-12	Valid Delay from Rising Clock Edge.....	94
10-13	Valid Delay from Rising Clock Edge.....	95
10-14	Setup and Hold Times	95
10-15	Float Delay	95
10-16	Output Enable Delay	96
10-17	Pulse Width.....	96
10-18	PCH Test Load	96
10-19	SPI/GSPI/eSPI Timings	100
10-20	Controller Link Receive Timings	101
10-21	Controller Link Receive Slew Rate	101
10-22	Maximum Acceptable Overshoot/Undershoot Waveform	104
11-1	PCH BGA Ball Map (Top View – Upper Left).....	106
11-2	PCH BGA Ball Map (Top View – Upper Right).....	107
11-3	PCH BGA Ball Map (Top View – Lower Left).....	108
11-4	PCH BGA Ball Map (Top View – Lower Right).....	109
17-1	Basic eSPI Protocol	137
17-2	eSPI Slave Request to PCH for PCH Temperature	140
17-3	PCH Response to eSPI Slave with PCH Temperature	141
17-4	eSPI Slave Request to PCH for PCH RTC Time	141
17-5	PCH Response to eSPI Slave with RTC Time.....	142
20-1	Data Transfer on the I ² C Bus.....	163
24-1	LPC Interface Diagram.....	192
25-1	Integrated Clock Controller (ICC) Diagram	197
25-2	PCH ICC XTAL Input Configurations	198
26-1	Supported PCI Express* Link Configurations	203
26-2	Generation of SERR# to Platform	208
27-1	Conceptual Diagram of SLP_LAN#	234
29-1	Flow for Port Enable/Device Present Bits	254
29-2	Serial Data transmitted over the SGPIO Interface.....	257
32-1	Flash Descriptor Regions.....	279
33-1	Platform Setup with Intel® Trace Hub	284
33-2	Platform Setup with DCI Connection	285
34-1	UART Serial Protocol.....	288
34-2	UART Receiver Serial Data Sample Points.....	289
37-1	Example of GSX Topology	303

Tables

1-1	PCH I/O Capabilities	27
1-1	Desktop PCH SKUs	27
1-2	Workstation / Server PCH SKUs	28
1-3	Mobile PCH SKUs.....	29
1-4	Desktop PCH HSIO Details.....	30
1-5	Workstation / Server PCH HSIO Details	31
1-6	Mobile PCH HSIO Details	32
2-1	PCH Device and Revision ID Table	34
4-1	PCI Devices and Functions.....	38
4-2	Fixed I/O Ranges Decoded by PCH.....	40
4-3	Variable I/O Decode Ranges	43
4-4	PCH Memory Decode Ranges (Processor Perspective).....	43
4-5	Boot Block Update Scheme.....	46
5-1	Event Transitions that Cause Messages	48



6-1	Legacy Replacement Routing.....	52
8-1	Power Rails Descriptions PCH-H.....	58
9-1	Pin Straps.....	60
10-1	PCH Absolute Power Rail Maximum and Minimum Ratings	64
10-2	PCH I _{CC3} with Integrated 1.8V VRM Mode OFF (Desktop SKUs)	64
10-3	PCH I _{CC3} with Integrated 1.8V VRM Mode OFF (H Mobile SKUs)	66
10-4	PCH I _{CC3} with Integrated 1.8V VRM Mode ON (Desktop SKUs)	67
10-5	PCH I _{CC3} with Integrated 1.8V VRM Mode ON (H Mobile SKUs)	68
10-6	PCH VCCPRIM_1P05 Adder per HSIO Lane	69
10-7	PCH VCCPRIM_1P8 Adder with and without CNVi.....	69
10-8	PCH VCCPRIM_3P3 Adder with and without CNVi.....	69
10-9	Single-Ended Signal DC Characteristics as Inputs or Outputs	70
10-10	Single-Ended Signal DC Characteristics as Inputs or Outputs	76
10-11	Differential Signals Characteristics	77
10-12	Power Rail Operational Voltage Range	79
10-13	PCI Express* Interface Timings	80
10-14	DDC Characteristics	83
10-15	DisplayPort* Hot-Plug Detect Interface	84
10-16	Clock Timings.....	84
10-17	USB 2.0 Timing	89
10-18	USB 3.1 Interface Transmit and Receiver Timings	90
10-19	SATA Interface Timings	91
10-20	SMBus Timings.....	92
10-21	I2C and SMLink Timing (Sheet 1 of 2)	92
10-22	Intel® High Definition Audio (Intel® HD Audio) Timing.....	93
10-23	DMIC Timing.....	94
10-24	LPC Timing (24 MHz)	94
10-25	Miscellaneous Timings	96
10-26	SPI Timings (20 MHz)	96
10-27	SPI Timings (33 MHz)	97
10-28	SPI Timings (50 MHz)	97
10-29	SPI Touch Timings (20 MHz)	98
10-30	SPI Touch Timings (33 MHz) 1.8V/3.3V	98
10-31	eSPI Timings (33 MHz).....	99
10-32	eSPI Timings (50 MHz).....	99
10-33	eSPI Timings (66 MHz).....	100
10-34	GSPI Timings (25 MHz)	100
10-35	Controller Link Receive Timings	101
10-36	UART Timings	102
10-37	I ² S Timings - Master Mode.....	102
10-38	I2S Timing - Slave Mode (non S0ix)	102
10-39	I2S Timing - Slave Mode (S0ix)	102
10-40	3.3V Overshoot/Undershoot Specifications	103
11-1	PCH BGA Ballout.....	110
12-1	Counter Operating Modes	119
13-1	Signal Descriptions	122
13-2	Integrated Pull-Ups and Pull-Downs	124
13-3	I/O Signal Planes and States	124
16-1	Digital Display Signals.....	132
17-1	eSPI Channels and Supported Transactions	138
17-2	eSPI Virtual Wires (VW).....	139
18-1	GPIO Group Summary.....	144
18-2	General Purpose I/O Signals.....	145
21-1	GbE LAN Signals.....	166
21-2	Integrated Pull-Ups and Pull-Downs	167



21-3	Power Plane and States for Output Signals	167
21-4	Power Plane and States for Input Signals.....	167
21-5	LAN Mode Support	171
22-1	Interrupt Options - 8259 Mode.....	173
22-2	Interrupt Options - APIC Mode	174
22-3	Interrupt Logic Signals.....	175
22-4	Interrupt Controllers PIC	176
22-5	Interrupt Status Registers	177
22-6	Content of Interrupt Vector Byte	177
22-7	Stop Frame Explanation	184
22-8	Data Frame Format	185
23-1	I/O Signal Planes and States.....	188
23-2	IPC Initiator -> Target flows	190
24-1	LPC Cycle Types Supported	194
24-2	Start Field Bit Definitions.....	194
24-3	Cycle Type Bit Definitions	194
24-4	Transfer Size Bit Definition	195
24-5	SYNC Bit Definition.....	195
25-1	PCH ICC Clocking Profile Support.....	198
25-1	Signal Descriptions.....	199
25-2	I/O Signal Pin States	199
26-1	Power Plane and States for PCI Express* Signals	202
26-2	PCI Express* Port Feature Details.....	202
26-3	MSI Versus PCI IRQ Actions.....	205
27-1	General Power States for Systems Using the PCH	216
27-2	State Transition Rules for the PCH	217
27-3	System Power Plane	218
27-4	Causes of SMI and SCI	219
27-5	Sleep Types	222
27-6	Causes of Wake Events	223
27-7	Transitions Due to Power Failure	224
27-8	Supported Deep Sx Policy Configurations.....	225
27-9	Deep Sx Wake Events.....	226
27-10	Transitions Due to Power Button	227
27-11	Write Only Registers with Read Paths in ALT Access Mode.....	231
27-12	PIC Reserved Bits Return Values	231
27-13	SUSPWRDNACK/SUSWARN#/GPP_A13 Pin Behavior	235
27-14	SUSPWRDNACK During Reset	235
27-15	Causes of Host and Global Resets.....	237
28-1	RTC Crystal Requirements	242
28-2	External Crystal Oscillator Requirements	242
29-1	Signals Description.....	243
31-1	I ² C Block Read.....	264
31-2	Enable for SMBALERT#	266
31-3	Enables for SMBus Slave Write and SMBus Host Events	267
31-4	Enables for the Host Notify Command	267
31-5	Slave Write Registers.....	268
31-6	Command Types	268
31-7	Slave Read Cycle Format.....	269
31-8	Data Values for Slave Read Registers.....	270
31-9	Host Notify Format	272
31-10	Slave Read Cycle Format.....	272
31-11	Data Values for Slave Read Registers.....	273
31-12	Enables for SMBus Slave Write and SMBus Host Events	274
32-1	Signal Description	275



32-2	SPI0 Flash Regions	278
32-3	Region Size Versus Erase Granularity of Flash Components	278
32-4	Region Access Control Table.....	280
33-1	Testability Signals.....	282
33-2	Power Planes and States for Testability Signals.....	283
35-1	Signal Descriptions	293
39-1	Private Configuration Space Register Target Port IDs.....	306
40-1	SD Working Modes.....	309



Revision History

Revision Number	Description	Revision Date
001	<ul style="list-style-type: none"> Initial Release 	April 2018
002	<p>Chapter 10, Electrical Characteristics</p> <ul style="list-style-type: none"> Added note 15 in Table 10-16. Updated Table 10-18 to include USB 3.1 Gen1(5 GT/s) and USB 3.1 Gen2(10 GT/s) spec. Updated signal names in Table 10-22 and Table 10-24. Added USB 2.0 undershoot/overshoot spec in Table 10-40. <p>Chapter 11, Ballout Definition</p> <ul style="list-style-type: none"> Updated information for x4 DMI only support. Removed [7:4] lane and x8 width details. <p>Chapter 24, Low Pin Count</p> <ul style="list-style-type: none"> Updated the LPC specification link in "References section" <p>Chapter 26, PCI Express</p> <ul style="list-style-type: none"> Added Precision Time Measurement (PTM) details <p>Chapter 33, Testability</p> <ul style="list-style-type: none"> Added a note in Section 33.3 stating that DCI and kernel level debugger are mutually exclusive <p>Chapter 38, Direct Media Interface</p> <ul style="list-style-type: none"> Updated information for x4 DMI only support. Removed [7:4] lane and x8 width details. <p>Chapter 39, Private Configuration Space Target Port ID</p> <ul style="list-style-type: none"> Changed chapter name to "Private Configuration Space Target Port ID". Updated Table 39.1. 	August 2018
003	<p>Chapter 2, PCH Controller Device IDs</p> <ul style="list-style-type: none"> Updated Section 2.1. Updated Device and Revision ID table. <p>Chapter 10, Electrical Characteristics</p> <ul style="list-style-type: none"> Updated VOH min Value in Table 10-9. Updated I²S timing specification in Table 10-31, Table 10-37 and Table 10-36. Updated Figure 10-7, Table 10-24, Table 10-26, Table 10-27, Table 10-28 and Table 10-34. Added Table 10-30. <p>Chapter 22, Low Pin Count</p> <ul style="list-style-type: none"> Updated the LPC specification link in Section 24.2. Updated the LPC Interface diagram. Updated signal description. Updated I/O Signal Planes and States. <p>Chapter 25, PCH and System Clocks</p> <ul style="list-style-type: none"> Included updated ICC diagrams which show HDA_PLL and XTAL Direct Paths for Display clock generation. Also removed conditional text for CPUNSSC clock descriptions. Added missing HM375, QM375, and CM248 details to PCH ICC Clocking. Profile Support table. <p>Chapter 26, PCI Express* (PCIe*)</p> <p>Corrected HM370 SKU PCIe* Controller Remapping Details.</p>	August 2018



Revision Number	Description	Revision Date
004	<p>Chapter 1, Introduction</p> <ul style="list-style-type: none"> Added Z390 details Added C242 and C246 SKU details <p>Chapter 2, PCH Controller Device IDs</p> <ul style="list-style-type: none"> Added C242 and C246 SKU details in Table 2-1 <p>Chapter 5, System Management</p> <ul style="list-style-type: none"> Removed "Second hard-coded timer timeout to generate reboot" from Section 5.3, "Features" Changed Intel ME to Intel® CSME <p>Chapter 10, Electrical and Thermal Characteristics</p> <ul style="list-style-type: none"> Updated Chapter title to "Electrical and Thermal Characteristics" Removed PCH Power Supply Range table Updated Section 10.2, "Thermal Specification" Updated Table 10-1, Table 10-2, Table 10-3, Table 10-4, Table 10-5, Table 10-6, Table 10-9, Table 10-11, Table 10-12, Table 10-14, Table 10-15, Table 10-16 Updated notes section of Table 10-10 Updated notes for t108 in Table 10-17 Updated Figure 10-11 SPI Clock frequency updated to 50 MHz from 48MHz in Table 10-28 SPI Touch Timings (17 MHz) table added Table 10-29 SPI Clock frequency updated to 25 MHz from 20MHz in Table 10-34 eSPI Timings added (30,48 and 60 MHz) (Table 10-31, Table 10-32, Table 10-33). <p>Chapter 21, Gigabit Ethernet Controller</p> <ul style="list-style-type: none"> Updated Section 21.2, "References" and Section 21.7, "Functional Description" <p>Chapter 24, Low Pin Count (LPC)</p> <ul style="list-style-type: none"> Updated Section 24.2, "References" and Section 24.7.6, "SYNC Error Indication" <p>Chapter 25, PCH and System Clocks</p> <ul style="list-style-type: none"> Added C242 and C246 SKU details in Table 25-1 Added the signal Type for XTAL_Freq_Select in Table 25-1 <p>Chapter 26, PCI Express (PCIe*)</p> <ul style="list-style-type: none"> Added Z390 SKU Figure 26-1 Removed HM375, QM375, and CM248 SKUs that are no longer supported SKUs <p>Chapter 28, Real Time Clock (RTC)</p> <ul style="list-style-type: none"> Updated Section 28.5.6 with correct design guide references Updated table note in Section 28.4 <p>Chapter 32, Serial Peripheral Interface (SPI)</p> <ul style="list-style-type: none"> Updated Section 32.6.1.3.2, "Flash Descriptor CPU Complex Soft Strap Section" Updated Table 32-4 Updated Section 32.5, "I/O Signal Planes and States" <p>Chapter 35, Universal Serial Bus (USB)</p> <ul style="list-style-type: none"> Added C242 and C246 SKU details in Figure 35-1 <p>Chapter 40, Secure Digital eXtended Capacity (SDXC)</p> <ul style="list-style-type: none"> Updated Section 40.4, "Signal Description" Updated Section 40.5, "I/O Signal Planes and States" 	December 2018



Revision Number	Description	Revision Date
005	<p>Chapter 10, Electrical and Thermal Characteristics</p> <ul style="list-style-type: none"> Updated Table 10-16 and Table 10-20. Added Table 10-21 and Table 10-23. <p>Chapter 17, Enhanced Serial Peripheral Interface (eSPI)</p> <ul style="list-style-type: none"> Removed information on PECI over eSPI. <p>Chapter 26, PCI Express (PCIe*)</p> <ul style="list-style-type: none"> Corrected QM370 SKU data table. <p>Chapter 35, Universal Serial Bus (USB)</p> <ul style="list-style-type: none"> Changed product name on the Supported USB 2.0 Ports table to generic name. Changed CNL/CFL to PCH-U/Y. <p>Chapter 36, Connectivity Integrated (CNVi)</p> <ul style="list-style-type: none"> Updated Section 36.4, "Signal Description" table. 	March 2019
006	<p>Chapter 1, Introduction</p> <ul style="list-style-type: none"> Updated no of PCIe Root Port for Base-U in Table 1-2. <p>Chapter 2, PCH Controller Device IDs</p> <ul style="list-style-type: none"> Updated Section 2.1 Updated Device and Revision ID table <p>Chapter 3, Flexible IO</p> <ul style="list-style-type: none"> Corrected USB naming <p>Chapter 10, Electrical and Thermal Characteristics</p> <ul style="list-style-type: none"> Updated Table 10-21 Updated T150(min) and T154(min) value in Table 10-24. <p>Chapter 36, Connectivity Integrated (CNVi)</p> <ul style="list-style-type: none"> Added a Note under signal description table 	September 2019
007	<p>Chapter 26, PCI Express (PCIe*)</p> <ul style="list-style-type: none"> Added a note in Section 26.4.13, "Precision Time Measurement (PTM)" 	July 2020
008	<p>Chapter 27, "Power Management"</p> <ul style="list-style-type: none"> Added note to SX_EXIT_HOLDOFF# signal description 	November 2020
009	<p>Chapter 33, "Testability"</p> <ul style="list-style-type: none"> Updated content in Table 33-1 Updated content in Table 33-2 <p>Chapter 39, "Private Configuration Space Target Port ID"</p> <ul style="list-style-type: none"> Updated DCI port ID in Table 39-1 	September 2021

§ §



1 Introduction

1.1 About this Manual

This document is intended for Original Equipment Manufacturers (OEMs), Original Design Manufacturers (ODM) and BIOS vendors creating products based on the Intel® 300 Series and Intel® C240 Series Chipset Families Platform Controller Hub (PCH).

Throughout this document, the Platform Controller Hub (PCH) is used as a general term and refers to all SKUs, unless specifically noted otherwise.

Throughout this document, PCH refers to desktop, and mobile segment PCH SKUs, unless specifically noted otherwise.

Throughout this document, the terms “Desktop” and “Desktop Only” refers to information that is applicable only to Desktop PCH, unless specifically noted otherwise.

Throughout this document, the terms “Mobile” and “Mobile Only” refers to information that is applicable only to Mobile PCH, unless specifically noted otherwise.

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. This manual abbreviates buses as B_n , devices as D_n and functions as F_n . For example Device 31 Function 0 is abbreviated as D31:F0, Bus 1 Device 8 Function 0 is abbreviated as B1:D8:F0. Generally, the bus number will not be used, and can be considered to be Bus 0.

1.2 Overview

The PCH provides extensive I/O support. Functions and capabilities include:

- ACPI Power Management Logic Support, Revision 4.0a
- PCI Express* Base Specification Revision 3.0
- Integrated Serial ATA Host Controller 3.2
- xHCI USB 3.1 Controller
- USB Dual Role Capability
- Direct Media Interface (DMI)
- Serial Peripheral Interface (SPI)
- Enhanced Serial Peripheral Interface (eSPI)
- Flexible I/O—Allows some high speed I/O signals to be configured as PCIe*, SATA 6Gb/s or USB 3.1
- General Purpose Input Output (GPIO)
- Low Pin Count (LPC) interface
- Interrupt Controller
- Timer functions
- System Management Bus (SMBus) Specification, Version 2.0
- Integrated Clock Controller (ICC) / Real Time Clock (RTC)



- Intel® High Definition Audio and Intel® Smart Sound Technology (Intel® SST) supporting Intel® HD Audio, I²S, SoundWire*, and DMIC interfaces
- Intel® Serial I/O UART Host Controllers
- Intel® Serial I/O I²C host Controllers
- Integrated 10/100/1000 Gigabit Ethernet MAC
- Integrated Sensor Hub (ISH)
- Supports Intel® Rapid Storage Technology (Intel® RST)
- Supports Intel® Active Management Technology (Intel® AMT)
- Supports Intel® Virtualization Technology for Directed I/O (Intel® VT-d)
- Supports Intel® Trusted Execution Technology (Intel® TXT)
- JTAG Boundary Scan support
- Intel® Trace Hub (Intel® TH) and Direct Connect Interface (DCI) for debug
- Supports Intel® Converged Security and Management Engine (Intel® CSME) with firmware version 12
- Integrated Intel® Wireless-AC Support Not all functions and capabilities may be available on all SKUs.

1.3 PCH SKUs

The following table provides an overview of the PCH I/O capabilities.

Table 1-1. Desktop PCH SKUs (Sheet 1 of 2)

Features	SKU				
	Q370	H370	B360	H310	Z390
DMI	DMI 3.0 x4	DMI 3.0 x4	DMI 3.0 x4	DMI 2.0 x4	DMI 3.0 x4
SATA 6 Gb/s Ports	Up to 6	Up to 6	Up to 6	Up to 4	Up to 6
PCIe	Up to 24 Gen 3 lanes	Up to 20 Gen 3 lanes	Up to 12 Gen 3 lanes	6 Gen 2 lanes	Up to 24 Gen 3 lanes
Total USB Ports (Maximum USB 3.1 Gen 1)	14 (10)	14 (8)	12 (6)	10 (4)	14 (10)
Maximum USB 3.1 Ports: Gen 2 (10 Gb/s) / Gen 1 (5 Gb/s)	6 / 10	4 / 8	4 / 6	0 / 4	6 / 10
Total USB 2.0 Ports	14	14	12	10	14
Intel® Smart Sound Technology	YES	YES	YES	NO	YES
Intel® CSME 12 Firmware	Corporate	Consumer / Corporate	Consumer / Corporate	Consumer	Consumer
Intel® AMT	YES	NO	NO	NO	NO
Intel® Optane™ Memory Support	YES	YES	YES	NO	YES



Table 1-1. Desktop PCH SKUs (Sheet 2 of 2)

Features	SKU				
	Q370	H370	B360	H310	Z390
Intel® Rapid Storage Technology	YES	YES	YES	YES	YES
Intel® RST for PCIe* Storage Ports ¹	3	2	1	0	3
Intel® RST RAID Support	YES	YES	NO	NO	YES
Integrated Intel® Wireless-AC Support	YES	YES	YES	YES	YES
eSPI Chip Select	1	1	1	1	2
Intel® Trusted Execution Technology	YES	NO	NO	NO	NO
Processor PCI Express 3.0 Lanes Configuration Support	1x16 or 2x8 or 1x8+2x4	1x16	1x16	1x16	1x16 or 2x8 or 1x8+2x4
Simultaneous Independent Display Support ²	3	3	3	2	3
Notes:					
1. Refer to Section 26.4.1 for more information on the Intel® RST for PCIe* Storage port locations.					
2. This is applicable to integrated graphics only.					

Table 1-2. Workstation / Server PCH SKUs (Sheet 1 of 2)

Features	SKU	
	C242	C246
DMI	DMI 3.0 x4	DMI 3.0 x4
SATA 6 Gb/s Ports	Up to 6	Up to 8
PCIe	Up to 10 Gen 3 lanes	Up to 24 Gen 3 lanes
Total USB Ports (Maximum USB 3.1 Gen 1)	12 (6)	14 (10)
Maximum USB 3.1 Ports: Gen 2 (10 Gb/s) / Gen 1 (5 Gb/s)	2 / 6	6 / 10
Total USB 2.0 Ports	12	14
Intel® Smart Sound Technology	Yes	Yes
Intel® CSME 12 Firmware	Corporate	Corporate
Intel® AMT	No	Yes
Intel® Optane™ Memory Support	No	Yes


Table 1-2. Workstation / Server PCH SKUs (Sheet 2 of 2)

Features	SKU	
	C242	C246
Intel® Rapid Storage Technology Enterprise	Yes	Yes
Intel® RST for PCIe* Storage Ports ¹	1	3
Integrated Intel® Wireless-AC Support	Yes	Yes
eSPI Chip Select	2	2
Intel® Trusted Execution Technology	Yes	Yes
Simultaneous Independent Display Support ²	No Integrated Graphics Support	3
Notes:		
1. Refer to Section 26.4.1 for more information on the Intel® RST for PCIe* Storage port locations.		
2. This is applicable to integrated graphics only.		

Table 1-3. Mobile PCH SKUs (Sheet 1 of 2)

Mobile PCH SKUs	SKU		
	HM370	QM370	CM246
DMI	DMI 3.0 x4	DMI 3.0 x4	DMI 3.0 x4
SATA 6 Gb/s Ports	4	4	8
PCIe*	Up to 16 Gen 3 lanes	Up to 20 Gen 3 lanes	Up to 24 Gen 3 lanes
Total USB Ports (Maximum USB 3.1 Gen 1)	14 (8)	14 (10)	14 (10)
Maximum USB 3.1 Ports: Gen 2 (10 Gb/s) / Gen 1 (5 Gb/s)	4 / 8	6 / 10	6 / 10
Total USB 2.0 Ports	14	14	14
Intel® Smart Sound Technology	Yes	Yes	Yes
Intel® CSME 12 Firmware	Consumer	Consumer / Corporate	Consumer / Corporate
Intel® AMT	No	Yes	Yes
Intel® Optane™ Memory Support	Yes	Yes	Yes
Intel® Rapid Storage Technology	Yes	Yes	Yes
Intel® RST for PCIe Storage Ports ¹	2	2	3
Intel® RST RAID Support	Yes	Yes	Yes
Integrated Intel® Wireless-AC Support	Yes	Yes	Yes
eSPI Chip Select	1	1	1
Intel® Trusted Execution Technology	No	Yes	Yes



Table 1-3. Mobile PCH SKUs (Sheet 2 of 2)

Mobile PCH SKUs	SKU		
	HM370	QM370	CM246
Simultaneous Independent Display Support ²	3	3	3
Notes: 1. Refer to Section 26.4.1 for more information on the Intel [®] RST for PCIe* Storage port locations. 2. This is applicable to integrated graphics only.			

Table 1-4. Desktop PCH HSIO Details (Sheet 1 of 2)

Flex I/O Lane	SKU				
	Q370	H370	B360	H310	Z390
0	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1/ Gen 2
1	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1/ Gen 2
2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1 / Gen 2
3	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1/ Gen 2
4	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1	N/A	USB 3.1 Gen 1/ Gen 2
5	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1	USB 3.1 Gen 1	N/A	USB 3.1 Gen 1/ Gen 2
6	USB 3.1 Gen 1, PCIe*	USB 3.1 Gen 1	N/A	N/A	USB 3.1 Gen 1, PCIe*
7	USB 3.1 Gen 1, PCIe*	USB 3.1 Gen 1	N/A	N/A	USB 3.1 Gen 1, PCIe*
8	USB 3.1 Gen 1, PCIe*	PCIe*	N/A	N/A	USB 3.1 Gen 1, PCIe*
9	USB 3.1 Gen 1, PCIe*	PCIe*	N/A	N/A	USB 3.1 Gen 1, PCIe*
10	PCIe*, GbE	PCIe*, GbE	PCIe*, GbE	PCIe*, GbE	PCIe*, GbE
11	PCIe*	PCIe*	PCIe*	PCIe*	PCIe*
12	PCIe*	PCIe*	PCIe*	PCIe*	PCIe*
13	PCIe*	PCIe*	PCIe*	PCIe*	PCIe*
14	PCIe*, GbE	PCIe*, GbE	PCIe*, GbE	GbE	PCIe*, GbE
15	PCIe*	PCIe*	PCIe*	N/A	PCIe*
16	PCIe*, SATA 0A	PCIe*, SATA 0A	PCIe*, SATA 0A	PCIe	PCIe*, SATA 0A
17	PCIe*, GbE, SATA 1A	PCIe*, GbE, SATA 1A	PCIe*, GbE, SATA 1A	PCIe, GbE	PCIe*, GbE, SATA 1A
18	PCIe*, GbE, SATA 0B	PCIe*, GbE, SATA 0B	GbE, SATA 0B	GbE, SATA 0	PCIe*, GbE, SATA 0B
19	PCIe*, SATA 1B	PCIe*, SATA 1B	SATA 1B	SATA 1	PCIe*, SATA 1B



Table 1-4. Desktop PCH HSIO Details (Sheet 2 of 2)

Flex I/O Lane	SKU				
	Q370	H370	B360	H310	Z390
20	PCIe*, SATA 2	PCIe*, SATA 2	SATA 2	SATA 2	PCIe*, SATA 2
21	PCIe*, SATA 3	PCIe*, SATA 3	SATA 3	SATA 3	PCIe*, SATA 3
22	PCIe*, SATA 4	SATA 4	SATA 4	N/A	PCIe*, SATA 4
23	PCIe*, SATA 5	SATA 5	SATA 5	N/A	PCIe*, SATA 5
24	PCIe*	PCIe*	N/A	N/A	PCIe*
25	PCIe*	PCIe*	N/A	N/A	PCIe*
26	PCIe*	PCIe*	PCIe*	N/A	PCIe*
27	PCIe*	PCIe*	PCIe*	N/A	PCIe*
28	PCIe*	PCIe*	PCIe*	N/A	PCIe*
29	PCIe*	PCIe*	PCIe*	N/A	PCIe*

Table 1-5. Workstation / Server PCH HSIO Details (Sheet 1 of 2)

Flex I/O Lane	SKU	
	C242	C246
0	USB 3.1 Gen 1/Gen 2	USB 3.1 Gen 1/Gen 2
1	USB 3.1 Gen 1/Gen 2	USB 3.1 Gen 1/Gen 2
2	USB 3.1 Gen 1	USB 3.1 Gen 1/Gen 2
3	USB 3.1 Gen 1	USB 3.1 Gen 1/Gen 2
4	USB 3.1 Gen 1	USB 3.1 Gen 1/Gen 2
5	USB 3.1 Gen 1	USB 3.1 Gen 1/Gen 2
6	PCIe*	USB 3.1 Gen 1,PCIe*
7	PCIe*	USB 3.1 Gen 1,PCIe*
8	N/A	USB 3.1 Gen 1,PCIe*
9	N/A	USB 3.1 Gen 1,PCIe*
10	GbE	PCIe*, GbE
11	N/A	PCIe*
12	N/A	PCIe*
13	N/A	PCIe*
14	PCIe*, GbE	PCIe*, GbE
15	PCIe	PCIe*
16	PCIe*, SATA 0A	PCIe*, SATA 0A
17	PCIe*, GbE, SATA 1A	PCIe*, GbE, SATA 1A
18	SATA 0B	PCIe*, GbE, SATA 0B
19	SATA 1B	PCIe*, SATA 1B
20	SATA 2	PCIe*, SATA 2
21	SATA 3	PCIe*, SATA 3



Table 1-5. Workstation / Server PCH HSIO Details (Sheet 2 of 2)

Flex I/O Lane	SKU	
	C242	C246
22	SATA 4	PCIe*, SATA 4
23	SATA 5	PCIe*, SATA 5
24	N/A	PCIe*, SATA 6
25	N/A	PCIe*, SATA 7
26	PCIe*	PCIe*
27	PCIe*	PCIe*
28	PCIe*	PCIe*
29	PCIe*	PCIe*

Table 1-6. Mobile PCH HSIO Details (Sheet 1 of 2)

Flex I/O Lane	SKU		
	HM370	QM370	CM246
0	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
1	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
3	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
4	USB 3.1 Gen 1	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
5	USB 3.1 Gen 1	USB 3.1 Gen 1/ Gen 2	USB 3.1 Gen 1/ Gen 2
6	USB 3.1 Gen 1	USB 3.1 Gen 1	USB 3.1 Gen 1, PCIe*
7	USB 3.1 Gen 1	USB 3.1 Gen 1	USB 3.1 Gen 1, PCIe*
8	N/A	USB 3.1 Gen 1	USB 3.1 Gen 1, PCIe*
9	N/A	USB 3.1 Gen 1	USB 3.1 Gen 1, PCIe*
10	GbE	PCIe, GbE	PCIe*, GbE
11	N/A	PCIe*	PCIe*
12	N/A	PCIe*	PCIe*
13	N/A	PCIe*	PCIe*
14	PCIe*, GbE	PCIe*, GbE	PCIe*, GbE
15	PCIe*	PCIe*	PCIe*
16	PCIe*, SATA 0A	PCIe*, SATA 0A	PCIe*, SATA 0A


Table 1-6. Mobile PCH HSIO Details (Sheet 2 of 2)

Flex I/O Lane	SKU		
	HM370	QM370	CM246
17	PCIe*, GbE, SATA 1A	PCIe*, GbE, SATA 1A	PCIe*, GbE, SATA 1A
18	PCIe*, GbE, SATA 0B	PCIe*, GbE, SATA 0B	PCIe*, GbE, SATA 0B
19	PCIe*, SATA 1B	PCIe*, SATA 1B	PCIe*, SATA 1B
20	PCIe*	PCIe*	PCIe*, SATA 2
21	PCIe*	PCIe*	PCIe*, SATA 3
22	PCIe*, SATA 4	PCIe*, SATA 4	PCIe*, SATA 4
23	PCIe*, SATA 5	PCIe*, SATA 5	PCIe*, SATA 5
24	PCIe*	PCIe*	PCIe*, SATA 6
25	PCIe*	PCIe*	PCIe*, SATA 7
26	PCIe*	PCIe*	PCIe*
27	PCIe*	PCIe*	PCIe*
28	PCIe*	PCIe*	PCIe*
29	PCIe*	PCIe*	PCIe*

§ §



2 PCH Controller Device IDs

2.1 Device and Revision ID Table

The Revision ID (RID) register is an 8-bit register located at offset 08h in the PCI header of every PCIe* function.

Table 2-1. PCH Device and Revision ID Table (Sheet 1 of 2)

Dev ID	Device Function - Device Description	B0 SRID	Notes
A300 - A31F	D31:F0 - LPC Controller (eSPI Enable Strap = 0)/ eSPI Controller (eSPI Enable Strap = 1)	10	PCH Device IDs: Q370: A306 H370: A304 Z390: A305 B360: A308 H310: A303 C242: A30A C246: A309 HM370: A30D QM370: A30C CM246: A30E
A320	D31:F1 - P2SB	10	
A321	D31:F2 - PMC	10	
A323	D31:F4 - SMBus	10	
A324	D31:F5 - SPI (Flash) Controller	10	
15BB	D31:F6 - GbE Controller	10	
15BC	D31:F6 - GbE Controller: Consumer	10	
A326	D31:F7 - Intel® Trace Hub	10	
A328	D30:F0 - UART #0	10	
A329	D30:F1 - UART #1	10	
A32A	D30:F2 - GSPI #0	10	
A32B	D30:F3 - GSPI #1	10	
A32C	D27:F4 - PCI Express* Root Port #21	F0	
A32D	D27:F5 - PCI Express* Root Port #22	F0	
A32E	D27:F6 - PCI Express* Root Port #23	F0	
A32F	D27:F7 - PCI Express* Root Port #24	F0	
A330	D29:F0 - PCI Express* Root Port #9	F0	
A331	D29:F1 - PCI Express* Root Port #10	F0	
A332	D29:F2 - PCI Express* Root Port #11	F0	
A333	D29:F3 - PCI Express* Root Port #12	F0	
A334	D29:F4 - PCI Express* Root Port #13	F0	
A335	D29:F5 - PCI Express* Root Port #14	F0	
A336	D29:F6 - PCI Express* Root Port #15	F0	
A337	D29:F7 - PCI Express* Root Port #16	F0	
A338	D28:F0 - PCI Express* Root Port #1	F0	
A339	D28:F1 - PCI Express* Root Port #2	F0	



Table 2-1. PCH Device and Revision ID Table (Sheet 2 of 2)

Dev ID	Device Function - Device Description	B0 SRID	Notes
A33A	D28:F2 - PCI Express* Root Port #3	F0	
A33B	D28:F3 - PCI Express* Root Port #4	F0	
A33C	D28:F4 - PCI Express* Root Port #5	F0	
A33D	D28:F5 - PCI Express* Root Port #6	F0	
A33E	D28:F6 - PCI Express* Root Port #7	F0	
A33F	D28:F7 - PCI Express* Root Port #8	F0	
A340	D27:F0 - PCI Express* Root Port #17	F0	
A341	D27:F1 - PCI Express* Root Port #18	F0	
A342	D27:F2 - PCI Express* Root Port #19	F0	
A343	D27:F3 - PCI Express* Root Port #20	F0	
A347	D25:F2 - UART #2	10	
A348	D31:F3 - cAVS	10	
A352	D23:F0 - SATA Controller (AHCI)	10	Desktop SKUs: Q370, H370, B360, H310, Z390, C242, C246.
A353	D23:F0 - SATA Controller (AHCI)	10	Mobile H SKUs: HM370, QM370, CM246.
A355	D23:F0 - SATA Controller (RAID 0/1/5/10)	10	Mobile H SKUs: HM370, QM370, CM246.
A356	D23:F0 - SATA Controller (RAID 0/1/5/10)	10	Desktop SKUs: Q370, H370, B360, H310, Z390, C242, C246.
A357	D23:F0 - SATA Controller (RAID 0/1/5/10)	10	Mobile H SKUs: HM370, QM370, CM246.
2822	D23:F0 - SATA Controller (RAID 0/1/5/10) - In-box Compatible ID (Intel® RST driver)	10	Desktop SKUs: Q370, H370, B360, H310, Z390, C242, C246.
282A	D23:F0 - SATA Controller (RAID 0/1/5/10) - In-box Compatible ID (Intel® RST driver)	10	Mobile H SKUs: HM370, QM370, CM246.
A35E	D23:F0 - SATA Controller Intel® Optane Memory	10	
A360	D22:F0 - HECI #1	10	
A361	D22:F1 - HECI #2	10	
A362	D22:F2 - IDE Redirection (IDER-R)	10	
A363	D22:F3 - Keyboard and Text (KT) Redirection	10	
A364	D22:F4 - HECI #3	10	
A365	D22:F5 - HECI #4	10	
A368	D21:F0 - I ² C Controller #0	10	
A369	D21:F1 - I ² C Controller #1	10	
A36A	D21:F2 - I ² C Controller #2	10	
A36B	D21:F3 - I ² C Controller #3	10	
A36D	D20:F0 - USB 3.1 xHCI	10	
A36E	D20:F1 - USB Device Controller (Dual Role)	10	
A36F	D20:F2 - Shared SRAM	10	
A370 - A373	D20:F3 - CNVi: Wi-Fi*	10	
A379	D18:F0 - Thermal Subsystem	10	
A37B	D18:F6 - SPI #2	10	
A37C	D19:F0 - Integrated Sensor Hub (ISH)	10	

§ §

3 Flexible I/O

3.1 Overview

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 Gigabit Ethernet (GbE) Controller, a PCIe* Controller, a Extensible Host Controller Interface (xHCI) USB 3.1 Controller, or a 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.

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

3.2 Flexible I/O Implementation

3.2.1 PCH-H

Figure 3-1. High Speed I/O (HSIO) Lane Multiplexing in PCH-H

Flex I/O Lane	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
High Speed I/O (HSIO) Type and Lane	USB3.1 Gen1/Gen2 #1	USB3.1 Gen1/Gen2 #2	USB3.1 Gen1/Gen2 #3	USB3.1 Gen1/Gen2 #4	USB3.1 Gen1/Gen2 #5	USB3.1 Gen1/Gen2 #6	USB3.1 Gen1 #7	USB3.1 Gen1 #8	USB3.1 Gen1 #9	USB3.1 Gen1 #10	PCIe* #5	PCIe* #6	PCIe* #7	PCIe* #8	PCIe* #9	PCIe* #10	PCIe* #11	PCIe* #12	PCIe* #13	PCIe* #14	PCIe* #15	PCIe* #16	PCIe* #17	PCIe* #18	PCIe* #19	PCIe* #20	PCIe* #21	PCIe* #22	PCIe* #23	PCIe* #24
Intel® RST Support	No Support						No Support		No Support		Yes		No Support		Yes		Yes													

The 30 HSIO lanes on PCH-H supports the following configurations:

1. Up to 24 PCIe* Lanes
 - A maximum of 16 PCIe* Ports (or devices) can be enabled
 - When a GbE Port is enabled, the maximum number of PCIe* Ports (or devices) that can be enabled reduces based off the following:
 Max PCIe* Ports (or devices) = 16 - GbE (0 or 1)
 - PCIe* Lanes 1-4 (PCIe* Controller #1), 5-8 (PCIe* Controller #2), 9-12 (PCIe* Controller #3), 13-16 (PCIe* Controller #4), 17-20 (PCIe* Controller #5), and 21-24 (PCIe* Controller #6) can be individually configured
2. Up to 10 SATA Lanes



- A maximum of 8 SATA Ports (or devices) can be enabled
 - SATA Lane 0 has the flexibility to be mapped to Flex I/O Lane 16 or 18
 - SATA Lane 1 has the flexibility to be mapped to Flex I/O Lane 17 or 19
3. Up to 6 USB 3.1 Gen 1/Gen 2 Lanes and 4 USB 3.1 Gen 1 Lanes
 - A maximum of 10 USB 3.1 Ports (or devices) can be enabled
 - USB 3.1 Gen 1 = 5 GT/s
 - USB 3.1 Gen 2 = 10 GT/s
 4. Up to 4 GbE Lanes
 - A maximum of 1 GbE Port (or device) can be enabled
 5. Supports up to 3 Remapped (Intel® Rapid Storage Technology) PCIe* storage devices
 - x2 and x4 PCIe* NVMe SSD
 - x2 and x4 Next Generation Intel® Optane™ Memory
 - Refer the "PCI Express* (PCIe*)" chapter for the PCH PCIe* Controllers, configurations, and lanes that can be used for Intel® Rapid Storage Technology PCIe* storage support
 6. For unused SATA/PCIe* Combo Lanes, Flex I/O Lanes that can be configured as PCIe* or SATA, the lanes must be statically assigned to SATA or PCIe* via the SATA/PCIe Combo Port Soft Straps .

3.3 Flexible I/O Lane Selection

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

3.3.1 PCIe*/SATA Lane Selection

In addition to static configuration via soft straps, Flexible I/O Lanes that have PCIe*/SATA multiplexing can be configured via SATAXPCIE signaling to support implementation like SATA Express or mSATA, where the port configuration is selected by the type of the add-in card that is used.



4 Memory Mapping

4.1 Overview

This section describes (from the processor perspective) the memory ranges that the PCH decodes.

4.2 Functional Description

4.2.1 PCI Devices and Functions

The PCH incorporates a variety of PCI devices and functions, as shown in the following table. If for some reason, the particular system platform does not want to support any one of the Device Functions, with the exception of D30:F0, they can individually be disabled. The integrated Gigabit Ethernet controller will be disabled if no Platform LAN Connect component is detected (refer [Chapter 21, "Gigabit Ethernet Controller"](#)).

When a function is disabled, it does not appear at all to the software. A disabled function will not respond to any register reads or writes, insuring that these devices appear hidden to software.

Table 4-1. PCI Devices and Functions (Sheet 1 of 3)

Devices and Function	Description
Bus M: Device 31: Function 0	LPC Interface (eSPI Enable Strap = 0) eSPI Interface (eSPI Enable Strap = 1)
Bus M: Device 31: Function 1	P2SB
Bus M: Device 31: Function 2	PMC
Bus M: Device 31: Function 3	cAVS (Audio, Voice, Speech)
Bus M: Device 31: Function 4	SMBus Controller
Bus M: Device 31: Function 5	SPI
Bus M: Device 31: Function 6	GbE Controller
Bus M: Device 31: Function 7	Intel® Trace Hub
Bus M: Device 30: Function 0	UART #0
Bus M: Device 30: Function 1	UART #1
Bus M: Device 30: Function 2	SPI #0
Bus M: Device 30: Function 3	SPI #1
Bus M: Device 29: Function 0	PCI Express* Port 9
Bus M: Device 29: Function 1	PCI Express* Port 10
Bus M: Device 29: Function 2	PCI Express* Port 11
Bus M: Device 29: Function 3	PCI Express* Port 12
Bus M: Device 29: Function 4	PCI Express* Port 13
Bus M: Device 29: Function 5	PCI Express* Port 14



Table 4-1. PCI Devices and Functions (Sheet 2 of 3)

Devices and Function	Description
Bus M: Device 29: Function 6	PCI Express* Port 15
Bus M: Device 29: Function 7	PCI Express* Port 16
Bus M: Device 28: Function 0	PCI Express* Port 1
Bus M: Device 28: Function 1	PCI Express* Port 2
Bus M: Device 28: Function 2	PCI Express* Port 3
Bus M: Device 28: Function 3	PCI Express* Port 4
Bus M: Device 28: Function 4	PCI Express* Port 5
Bus M: Device 28: Function 5	PCI Express* Port 6
Bus M: Device 28: Function 6	PCI Express* Port 7
Bus M: Device 28: Function 7	PCI Express* Port 8
Bus M: Device 27: Function 0	PCI Express* Port 17
Bus M: Device 27: Function 1	PCI Express* Port 18
Bus M: Device 27: Function 2	PCI Express* Port 19
Bus M: Device 27: Function 3	PCI Express* Port 20
Bus M: Device 27: Function 4	PCI Express* Port 21
Bus M: Device 27: Function 5	PCI Express* Port 22
Bus M: Device 27: Function 6	PCI Express* Port 23
Bus M: Device 27: Function 7	PCI Express* Port 24
Bus M: Device 25: Function 2	UART Controller #2 (Note: ACPI Device)
Bus M: Device 24: Function 0	Reserved. Used by the NVM Remapping.
Bus M: Device 23: Function 0	SATA Controller
Bus M: Device 22: Function 0	Intel® MEI #1
Bus M: Device 22: Function 1	Intel® MEI #2
Bus M: Device 22: Function 2	IDE-Redirection (IDE-R)
Bus M: Device 22: Function 3	Keyboard and Text (KT) Redirection
Bus M: Device 22: Function 4	Intel® MEI #3
Bus M: Device 22: Function 5	Intel® MEI #4
Bus M: Device 22: Function 7	WLAN
Bus M: Device 21: Function 0	I ² C Controller #0
Bus M: Device 21: Function 1	I ² C Controller #1
Bus M: Device 21: Function 2	I ² C Controller #2
Bus M: Device 21: Function 3	I ² C Controller #3
Bus M: Device 20: Function 0	xHCI Controller
Bus M: Device 20: Function 1	USB Device Controller
Bus M: Device 20: Function 2	Shared SRAM
Bus M: Device 20: Function 3	CNVi: Wi-Fi



Table 4-1. PCI Devices and Functions (Sheet 3 of 3)

Devices and Function	Description
Bus M: Device 20: Function 5	SDXC
Bus M: Device 19: Function 0	Integrated Sensor Hub
Bus M: Device 18: Function 0	Thermal Subsystem
Bus M: Device 18: Function 2	PMT
Bus M: Device 18: Function 6	SPI #2

4.2.2 Fixed I/O Address Ranges

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

Note:

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

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

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

Table 4-2. Fixed I/O Ranges Decoded by PCH (Sheet 1 of 3)

I/O Address	Read Target	Write Target	Internal Unit (unless[E]: External) ²	Separate Enable/Disable
20h – 21h	Interrupt Controller	Interrupt Controller	Interrupt	None
24h – 25h	Interrupt Controller	Interrupt Controller	Interrupt	None
28h – 29h	Interrupt Controller	Interrupt Controller	Interrupt	None
2Ch – 2Dh	Interrupt Controller	Interrupt Controller	Interrupt	None
2Eh-2Fh	Super I/O	Super I/O	[E] Forwarded to LPC/eSPI	Yes. IOE.SE
30h – 31h	Interrupt Controller	Interrupt Controller	Interrupt	None
34h – 35h	Interrupt Controller	Interrupt Controller	Interrupt	None
38h – 39h	Interrupt Controller	Interrupt Controller	Interrupt	None
3Ch – 3Dh	Interrupt Controller	Interrupt Controller	Interrupt	None
40h	Timer/Counter	Timer/Counter	8254 Timer	None
42h-43h	Timer/Counter	Timer/Counter	8254 Timer	None
4Eh-4Fh	Microcontroller	Microcontroller	[E] Forwarded to LPC/eSPI	Yes. IOE.ME2
50h	Timer/Counter	Timer/Counter	8254 Timer	None
52h-53h	Timer/Counter	Timer/Counter	8254 Timer	None
60h	Keyboard Controller	Keyboard Controller	[E] Forwarded to LPC/eSPI	Yes, with 64h. IOE.KE
61h	NMI Controller	NMI Controller	CPU I/F	None



Table 4-2. Fixed I/O Ranges Decoded by PCH (Sheet 2 of 3)

I/O Address	Read Target	Write Target	Internal Unit (unless[E]: External) ²	Separate Enable/Disable
62h	Microcontroller	Microcontroller	[E] Forwarded to LPC/eSPI	Yes, with 66h. IOE.ME1
63h	NMI Controller ¹	NMI Controller ¹	CPU I/F	Yes, alias to 61h. GIC.P61AE
64h	Keyboard Controller	Keyboard Controller	[E] Forwarded to LPC/eSPI	Yes, with 60h. IOE.KE
65h	NMI Controller ¹	NMI Controller ¹	CPU I/F	Yes, alias to 61h. GIC.P61AE
66h	Microcontroller	Microcontroller	[E] Forwarded to LPC/eSPI	Yes, with 62h. IOE.ME1
67h	NMI Controller ¹	NMI Controller ¹	CPU I/F	Yes, alias to 61h. GIC.P61AE
70h	RTC Controller	NMI and RTC Controller	RTC	None
71h	RTC Controller	RTC Controller	RTC	None
72h	RTC Controller	RTC Controller	RTC	None. Alias to 70h if RC.UE=0, else 72h
73h	RTC Controller	RTC Controller	RTC	None. Alias to 71h if RC.UE='0', else 73h
74h	RTC Controller	RTC Controller	RTC	None
75h	RTC Controller	RTC Controller	RTC	None
76h-77h	RTC Controller	RTC Controller	RTC	None. Alias to 70h-71h if RC.UE=0, else 76h-77h
80h ³	LPC/eSPI or PCIe*	LPC/eSPI or PCIe*	Read: LPC/[E] eSPI or PCIe Write: [E] LPC/[E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else LPC/eSPI
84h - 86h	LPC/eSPI or PCIe*	LPC/eSPI or PCIe*	Read: LPC/[E] eSPI or PCIe Write: [E] LPC/[E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else LPC/eSPI
88h	LPC/eSPI or PCIe*	LPC/eSPI or PCIe*	Read: LPC/[E] eSPI or PCIe Write: [E] LPC/[E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else LPC/eSPI
8Ch - 8Eh	LPC/eSPI or PCIe*	LPC/eSPI or PCIe*	Read: LPC/[E] eSPI or PCIe Write: [E] LPC/[E] eSPI or [E] PCIe	None. PCIe if GCS.RPR='1', else LPC/eSPI



Table 4-2. Fixed I/O Ranges Decoded by PCH (Sheet 3 of 3)

I/O Address	Read Target	Write Target	Internal Unit (unless[E]: External) ²	Separate Enable/Disable
90h	LPC/eSPI	LPC/eSPI	Read: LPC/[E] eSPI Write: [E] LPC/[E] eSPI	None. Alias to 80h
92h	Reset Generator	Reset Generator	CPU I/F	None
94h - 96h	LPC/eSPI	LPC/eSPI	Read: LPC/[E] eSPI Write: [E] LPC/[E] eSPI	None. Alias to 8xh
98h	LPC/eSPI	LPC/eSPI	Read: LPC/[E] eSPI Write: [E] LPC/[E] eSPI	None. Alias to 88h
9Ch - 9Eh	LPC/eSPI	LPC/eSPI	Read: LPC/[E] eSPI Write: [E] LPC/[E] eSPI	None. Alias to 8xh
A0h - A1h	Interrupt Controller	Interrupt Controller	Interrupt	None
A4h - A5h	Interrupt Controller	Interrupt Controller	Interrupt	None
A8h - A9h	Interrupt Controller	Interrupt Controller	Interrupt	None
ACh - ADh	Interrupt Controller	Interrupt Controller	Interrupt	None
B0h - B1h	Interrupt Controller	Interrupt Controller	Interrupt	None
B2h - B3h	Power Management	Power Management	Power Management	None
B4h - B5h	Interrupt Controller	Interrupt Controller	Interrupt	None
B8h - B9h	Interrupt Controller	Interrupt Controller	Interrupt	None
BCh - BDh	Interrupt Controller	Interrupt Controller	Interrupt	None
200h-207h	Gameport Low	Gameport Low	[E] Forwarded to LPC/eSPI	Yes. IOE.LGE
208h-20Fh	Gameport High	Gameport High	[E] Forwarded to LPC/eSPI	Yes. IOE.HGE
4D0h - 4D1h	Interrupt Controller	Interrupt Controller	Interrupt Controller	None
CF9h	Reset Generator	Reset Generator	Interrupt controller	None
Notes:				
1. Only if the Port 61 Alias Enable bit (GIC.P61AE) bit is set. Otherwise, the cycle is internally terminated by the PCH.				
2. Destination of LPC/eSPI is depending on the eSPI/LPC Pin Strap, '1': eSPI, '0': LPC.				
3. This includes byte, word or double-word (DW) access at I/O address 80h.				

4.2.3 Variable I/O Decode Ranges

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

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



Table 4-3. Variable I/O Decode Ranges

Range Name ¹	Mappable	Size (Bytes)	Target
ACPI	Anywhere in 64K I/O Space	256	Power Management
IDE Bus Master	Anywhere in 64K I/O Space	16 or 32 Bytes	Intel® AMT IDE-R
SMBus	Anywhere in 64K I/O Space	32	SMB Unit
TCO	Anywhere in 64K I/O Space	32	SMB Unit
Parallel Port	3 ranges in 64K I/O Space	8	LPC/eSPI
Serial Port 1	8 Ranges in 64K I/O Space	8	LPC/eSPI
Serial Port 2	8 Ranges in 64K I/O Space	8	LPC/eSPI
Serial Port 3	8 Ranges in 64K I/O space	8	LPC/eSPI
Floppy Disk Controller	2 Ranges in 64K I/O Space	8	LPC/eSPI
LPC Generic 1	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 2	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 3	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 4	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
IO Trapping Ranges	Anywhere in 64K I/O Space	1 to 256 Bytes	Trap
Serial ATA Index/Data Pair	Anywhere in 64K I/O Space	16	SATA Host Controller
PCI Express Root Ports	Anywhere in 64K I/O Space	I/O Base/Limit	PCI Express* Root Ports
Keyboard and Text (KT)	Anywhere in 64K I/O Space	8	Intel® AMT Keyboard and Text
Notes:			
1. All ranges are decoded directly from DMI.			

4.3 Memory Map

Below table shows (from the Processor perspective) the memory ranges that the PCH will decode. Cycles that arrive from DMI that are not directed to any of the internal memory targets that decode directly from DMI will be master aborted.

PCIe* cycles generated by external PCIe* masters will be positively decoded unless they fall in the PCI-PCI bridge memory forwarding ranges (those addresses are reserved for PCI peer-to-peer traffic). If the cycle is not in the internal LAN controller's range, it will be forwarded up to DMI. Software must not attempt locks to the PCH's memory-mapped I/O ranges.

Note: Total ports are different for the different SKUs.

Table 4-4. PCH Memory Decode Ranges (Processor Perspective) (Sheet 1 of 3)

Memory Range	Target	Dependency/Comments
000E 0000h - 000E FFFFh	LPC/eSPI or SPI	Bit 6 in BIOS Decode Enable Register is set.
000F 0000h - 000F FFFFh	LPC/eSPI or SPI	Bit 7 in BIOS Decode Enable Register is set.
FECX X000h - FECX X040h	I/O(x)APIC inside PCH	XX controlled via APIC Range Select (ASEL) field and APIC Enable (AEN) bit.
FECX X000h - FECX XFFFh	PCIe port N (N=1 to 24)	X controlled via PCIe root port N IOxAPIC Range Base/Limit registers and Port N I/OxApic Enable (PAE) is set



Table 4-4. PCH Memory Decode Ranges (Processor Perspective) (Sheet 2 of 3)

Memory Range	Target	Dependency/Comments
FEF0 0000h - FFFF FFFFh	LPC/eSPI or SPI	uCode Patch Region Enable UCPR.UPRE is set
FFC0 0000h - FFC7 FFFFh FF80 0000h - FF87 FFFFh	LPC/eSPI or SPI	Bit 8 in BIOS Decode Enable Register is set
FFC8 0000h - FFCF FFFFh FF88 0000h - FF8F FFFFh	LPC/eSPI or SPI	Bit 9 in BIOS Decode Enable Register is set
FFD0 0000h - FFD7 FFFFh FF90 0000h - FF97 FFFFh	LPC/eSPI or SPI	Bit 10 in BIOS Decode Enable Register is set
FFD8 0000h - FDDF FFFFh FF98 0000h - FF9F FFFFh	LPC/eSPI or SPI	Bit 11 in BIOS Decode Enable Register is set
FFE0 0000h - FFE7 FFFFh FFA0 0000h - FFA7 FFFFh	LPC/eSPI or SPI	Bit 12 in BIOS Decode Enable Register is set Refer to Table 4-5 - Boot Block Update Scheme for swappable ranges
FFE8 0000h - FFEF FFFFh FFA8 0000h - FFAF FFFFh	LPC/eSPI or SPI	Bit 13 in BIOS Decode Enable Register is set Refer to Table 4-5 - Boot Block Update Scheme for swappable ranges
FFF0 0000h - FFF7 FFFFh FFB0 0000h - FFB7 FFFFh	LPC/eSPI or SPI	Bit 14 in BIOS Decode Enable Register is set Refer to Table 4-5 - Boot Block Update Scheme for swappable ranges
FFFC 0000h - FFFF FFFFh	LPC/eSPI or SPI	Always enabled. Refer to Table 4-5 - Boot Block Update Scheme for swappable ranges
FFF8 0000h - FFFB FFFFh FFB8 0000h - FFBF FFFFh	LPC/eSPI or SPI	Always enabled. Refer to Table 4-5 - Boot Block Update Scheme for swappable ranges
FF70 0000h - FF7F FFFFh FF30 0000h - FF3F FFFFh	LPC/eSPI or SPI	Bit 3 in BIOS Decode Enable Register is set
FF60 0000h - FF6F FFFFh FF20 0000h - FF2F FFFFh	LPC/eSPI or SPI	Bit 2 in BIOS Decode Enable Register is set
FF50 0000h - FF5F FFFFh FF10 0000h - FF1F FFFFh	LPC/eSPI or SPI	Bit 1 in BIOS Decode Enable Register is set
FF40 0000h - FF4F FFFFh FF00 0000h - FF0F FFFFh	LPC/eSPI or SPI	Bit 0 in BIOS Decode Enable Register is set
FED0 X000h - FED0 X3FFh	HPET	BIOS determines "fixed" location which is one of four 1 KB ranges where X (in the first column) is 0h, 1h, 2h, or 3h.
FED4 0000h - FED4 7FFFh	LPC or SPI	TPM and Trusted Mobile KBC
FED4 C000h - FED4 FFFFh	PCH Internal (PSF Error Handler)	Always enabled
FED5 0000h - FED5 FFFFh	Intel® ME	Always enabled
FED7 0000h - FED7 4FFFh	Internal Device	Security feature related
128 KB anywhere in 4 GB range	LAN Controller (CSR registers)	Enable via standard PCI mechanism (Device 31:Function 6)
4 KB anywhere in 4 GB range	LAN Controller (LAN space on Flash)	Enable via standard PCI mechanism (Device 31:Function 6)
64 KB anywhere in 64-bit address range	USB 3.1 Host Controller	Enable via standard PCI mechanism (Device 20, Function 0)
2 MB anywhere in 4 GB range	USB Device Controller	Enable via standard PCI mechanism (Device 20, Function 1)
24 KB anywhere in 4 GB range	USB Device Controller	Enable via standard PCI mechanism (Device 20, Function 1)


Table 4-4. PCH Memory Decode Ranges (Processor Perspective) (Sheet 3 of 3)

Memory Range	Target	Dependency/Comments
16 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
4 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
64 KB anywhere in 64-bit addressing space	Intel® HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
64 KB anywhere in 4 GB range	LPC/eSPI	LPC Generic Memory Range. Enable via setting bit[0] of the LPC Generic Memory Range register (D31:F0:offset 98h).
32 Bytes anywhere in 64-bit address range	SMBus	Enable via standard PCI mechanism (Device 31: Function 4)
2 KB anywhere above 64 KB to 4 GB range	SATA Host Controller	AHCI memory-mapped registers. Enable via standard PCI mechanism (Device 23: Function 0)
Memory Base/Limit anywhere in 4 GB range	PCI Express Root Ports	Enable via standard PCI mechanism
Prefetchable Memory Base/Limit anywhere in 64-bit address range	PCI Express Root Ports	Enable via standard PCI mechanism
4 KB anywhere in 64-bit address range	Thermal Reporting	Enable via standard PCI mechanism (Device 18: Function 0)
16 Bytes anywhere in 64-bit address range	Intel® MEI #1, #2, #3, #4	Enable via standard PCI mechanism
4 KB anywhere in 4 GB range	Intel® AMT Keyboard and Text	Enable via standard PCI mechanism (Device 22: Function 3)
Eight 4 KB slots anywhere in 64-bit address range	UART, GPI and I ² C controllers	Enable via standard PCI mechanism
4 KB slots anywhere in 64-bit address range	SDXC controllers	Enable via standard PCI mechanism
1 MB (BAR0) or 4 KB (BAR1) in 4 GB range	Integrated Sensor Hub	Enable via standard PCI mechanism (Device 19: Function 0)
8 KB slot anywhere in 4 GB range	Integrated Wi-Fi*	Enable via standard PCI mechanism

4.3.1 Boot Block Update Scheme

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

For FWH when top swap is enabled, accesses to FFFF 0000h-FFFF FFFFh are directed to FFFE 0000h-FFFE FFFFh and vice versa. When the Top Swap Enable bit is 0, the PCH will not invert A16.

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



Table 4-5. Boot Block Update Scheme

BOOT_BLOCK_SIZE Value	Accesses to	Being Directed to
000 (64 KB)	FFFF 0000h - FFFF FFFFh	FFFE 000h - FFFE FFFFh and vice versa
001 (128 KB)	FFFE 0000h - FFFF FFFFh	FFFC 0000h - FFFD FFFFh and vice versa
010 (256 KB)	FFFC 0000h - FFFF FFFFh	FFF8 0000h - FFFB FFFFh and vice versa
011 (512 KB)	FFF8 0000h - FFFF FFFFh	FFF0 0000h - FFF7 FFFFh and vice versa
100 (1 MB)	FFF0 0000h - FFFF FFFFh	FFE0 0000h - FFEF FFFFh and vice versa
101 - 111	Reserved	Reserved
Note: This bit is automatically set to 0 by RTCRST#, but not by PLTRST#.		

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

The algorithm is:

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

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

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

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





5 System Management

5.1 Acronyms

Acronyms	Description
BMC	Baseboard Management Controller
SPD	Serial Presence Detect
TCO	Total Cost of Ownership

5.2 Overview

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.

5.3 Features

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
- Various Error detection (such as ECC Errors) indicated by host controller:
 - Can generate SMI#, SCI, SERR, SMI, or TCO interrupt
- Intruder Detect input:
 - Can generate TCO interrupt or SMI#

5.3.1 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.

5.3.1.1 Handling an Intruder

The PCH has an input signal, INTRUDER#, that can be attached to a switch that is activated by the system's case being open. This input has a two RTC clock debounce. If INTRUDER# goes active (after the debouncer), this will set the INTRD_DET bit in the TCO2_STS register. The INTRD_SEL bits in the TCO_CNT register can enable the PCH to cause an SMI# or interrupt. The BIOS or interrupt handler can then cause a transition to the S5 state by writing to the SLP_EN bit.

The software can also directly read the status of the INTRUDER# signal (high or low) by clearing and then reading the INTRD_DET bit. This allows the signal to be used as a GPI if the intruder function is not required.

If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET bit will go to a 0 when INTRUDER# input signal goes inactive.

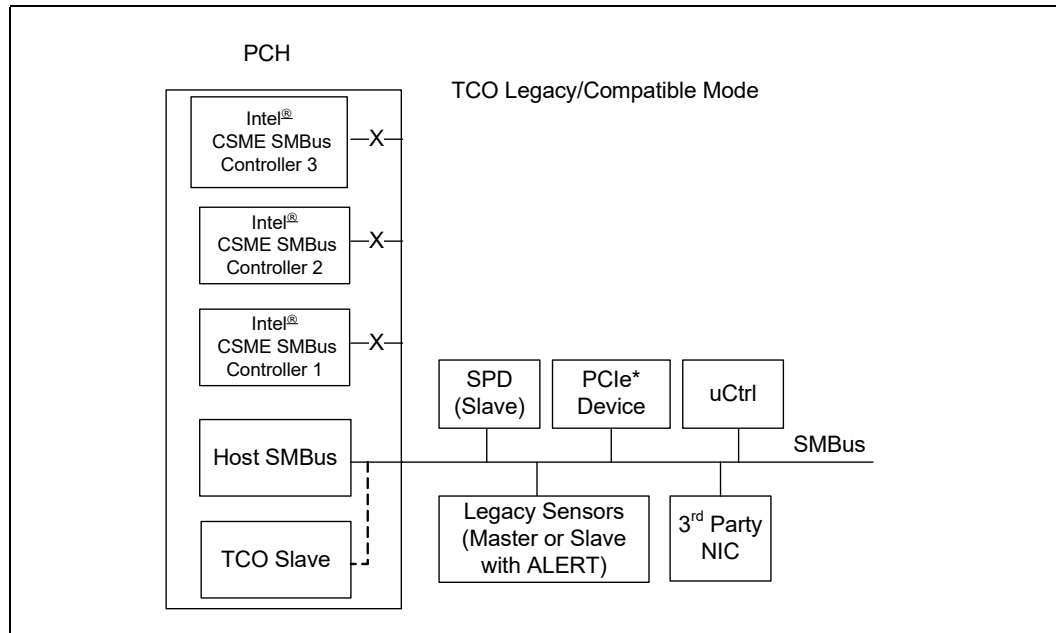
Note: This is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

5.3.2 TCO Modes

5.3.2.1 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. In this mode, the Intel® Converged Security and Management Engine (Intel® CSME) SMBus controllers are not used and should be disabled by soft strap.

Figure 5-1. 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 5-1. Event Transitions that Cause Messages (Sheet 1 of 2)

Event	Assertion	Deassertion	Comments
INTRUDER# pin	Yes	No	Must be in "hung S0" state
Watchdog Timer Expired	Yes	NA	"Hung S0" state entered
SMBALERT# pin	Yes	Yes	Must be in "Hung S0" state



Table 5-1. Event Transitions that Cause Messages (Sheet 2 of 2)

Event	Assertion	Deassertion	Comments
BATLOW#	Yes	Yes	Must be in "Hung S0" state
CPU_PWR_FLR	Yes	No	"Hung S0" state entered

5.3.2.2 Advanced TCO Mode

The PCH supports the Advanced TCO mode in which SMLink0 and SMLink1 are used in addition to the host SMBus.

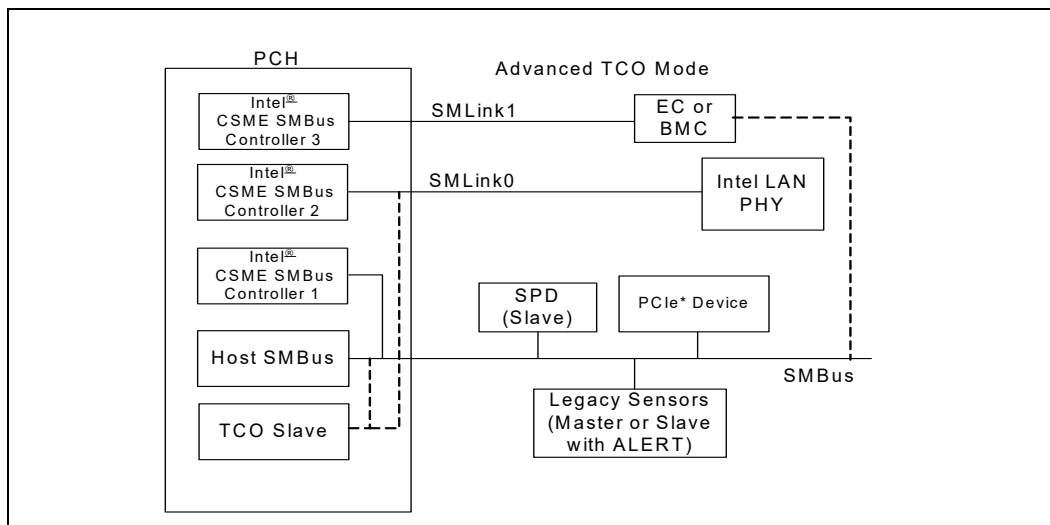
In this mode, the Intel® CSME SMBus controllers must be enabled by soft strap in the flash descriptor. Refer below figure for more details.

In advanced TCO mode, the TCO slave can either be connected to the host SMBus or the SMLink0.

SMLink0 is targeted for integrated LAN use. When an Intel LAN PHY is connected to SMLink0, a soft strap must be set to indicate that the PHY is connected to SMLink0. When the Fast Mode is enabled using a soft strap, the interface will be running at the frequency of up to 1 MHz depending on different factors such as board routing or bus loading.

SMLink1 can be connected to an Embedded Controller (EC) or Baseboard Management Controller (BMC) use. In the case where a BMC is connected to SMLink1, the BMC communicates with the Intel® Converged Security and Management Engine through the Intel® CSME SMBus connected to SMLink1. The host and TCO slave communicate with BMC through SMBus.

Figure 5-2. Advanced TCO Mode



§ §



6 High Precision Event Timer (HPET)

6.1 Overview

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.

6.1.1 Timer Accuracy

The timers are accurate over any 1 ms period to within 0.05% of the time specified in the timer resolution fields.

Within any 100-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 24 MHz XTAL as its clock. The accuracy of the main counter is as accurate as the crystal that is used in the system. The XTAL clock frequency is determined by the pin strap that is sampled on RSMRST#.

6.1.2 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 24 MHz 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 24 MHz clock during boot time to an accuracy that ensures the error introduced by this off-load is less than 10 ppb (.000001%).

When the 24 MHz clock is active, the 64-bit counter will increment by one each cycle of the 24 MHz clock when enabled. When the 24 MHz 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 24 MHz 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 24 MHz clock to shut down when it has no active comparators.

6.1.2.1 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.



6.1.3 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.

6.1.3.1 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 Table 6-1.

Table 6-1. 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.			

6.1.3.2 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.



6.1.3.3 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.

6.1.4 Periodic Versus Non-Periodic Modes

6.1.4.1 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.



6.1.4.2 Periodic Mode

When a timer is set up for periodic mode, the software writes a value in the timer's comparator value register. When the main counter value matches the value in the timer's comparator value register, an interrupt can be generated. The hardware will then automatically increase the value in the comparator value register by the last value written to that register.

To make the periodic mode work properly, the main counter is typically written with a value of 0 so that the first interrupt occurs at the right point for the comparator. If the main counter is not set to 0, interrupts may not occur as expected.

During run-time, the value in the timer's comparator value register can be read by software to find out when the next periodic interrupt will be generated (not the rate at which it generates interrupts). Software is expected to remember the last value written to the comparator's value register (the rate at which interrupts are generated).

If software wants to change the periodic rate, it should write a new value to the comparator value register. At the point when the timer's comparator indicates a match, this new value will be added to derive the next matching point.

If the software resets the main counter, the value in the comparator's value register needs to reset as well. This can be done by setting the **TIMERn_VAL_SET_CNF** bit. Again, to avoid race conditions, this should be done with the main counter halted. The following usage model is expected:

1. Software clears the ENABLE_CNF bit to prevent any interrupts.
2. Software Clears the main counter by writing a value of 00h to it.
3. Software sets the TIMER0_VAL_SET_CNF bit.
4. Software writes the new value in the TIMER0_COMPARATOR_VAL register.

Software sets the ENABLE_CNF bit to enable interrupts.

Note: As the timer period approaches zero, the interrupts associated with the periodic timer may not get completely serviced before the next timer match occurs. Interrupts may get lost and/or system performance may be degraded in this case.

Each timer is NOT required to support the periodic mode of operation. A capabilities bit indicates if the particular timer supports periodic mode. The reason for this is that supporting the periodic mode adds a significant amount of gates.

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

6.1.5 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.

6.1.6 Interrupt Levels

Interrupts directed to the internal 8259s are active high. Refer [Section 22.7, “Advanced Programmable Interrupt Controller \(APIC\) \(D31:F0\)”](#) for information regarding the polarity programming of the I/O APIC for detecting internal interrupts.

If the interrupts are mapped to the 8259 or I/O APIC and set for level-triggered mode, they can be shared with legacy interrupts. They may be shared although it is unlikely for the operating system to attempt to do this.

If more than one timer is configured to share the same IRQ (using the `TIMERn_INT_ROUT_CNF` fields), then the software must configure the timers to level-triggered mode. Edge-triggered interrupts cannot be shared.

6.1.7 Handling Interrupts

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

6.1.8 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.



7 Thermal Management

7.1 PCH Thermal Sensor

The PCH incorporates an on-die Digital Thermal Sensor (DTS) for thermal management.

7.1.1 Modes of Operation

The DTS has two usages when enabled:

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

7.1.2 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.

7.1.3 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.

7.1.4 Thermal Reporting to an EC

To support a platform EC that is managing the system thermals, the PCH provides the ability for the EC to read the PCH temperature over SMLink1 or over eSPI interface.

The PCH supports temperature reporting over SMLink1 using the General Purpose Block Read (GPBR). To enable the support, the related soft straps, SMLink1 GP Target Address and SMLink1 GP Target Address Enabled, must be set appropriately. The EC needs to issue an SMBus Block Read Command (0x40h) and receives a single byte of data indicating the PCH temperature. Upon reset, the value driven to the EC will be FFh. This indicates that BIOS has not enabled the sensor yet and the EC can assume



that the system is in the boot phase with unknown temperature. After the sensor is enabled, the EC will receive a value between 0h and 7Fh (0 °C to 127 °C). If the EC ever sees a value between 80h and FEh, that indicates an error has occurred, since the PCH should have shut down the platform before the temperature ever reached 128 °C (Catastrophic trip point will be below 128 °C). The PCH itself does not monitor the temperature and will not flag any error on the temperature value. The polling rate is recommended no more than once every 200ms, although EC can read as often as it wishes.

The PCH also supports temperature reporting over eSPI using eSPI OOB channel. For getting the PCH temperature over eSPI, the EC can send a request packet via OOB channel and receive a response packet consisting of 1 byte of PCH temperature. Refer to the Enhanced Serial Peripheral Interface (eSPI) chapter for detailed info on eSPI request and response packets for getting PCH temperature data.

7.1.5 Thermal Trip Signal (PCHHOT#)

The PCH provides PCHHOT# 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# 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 1 second.





8 Power and Ground Signals

This section describes the power rails and ground signals on the PCH.

Note: The historical Core well (on in S0 only) and ASW well (on in S0/M0 and Sx/M3) is no longer needed on the PCH due to several internal power management capabilities. The Primary well is equivalent to the historical Suspend well such that the supply is on in S0, S3, S4, S5. Refer to the Power Management Chapter for more details.

Table 8-1. Power Rails Descriptions PCH-H (Sheet 1 of 2)

Name	Description
VCCA_BCLK_1P05	Analog supply for BCLK circuitries: 1.05V
VCCA_SRC_1P05	Analog supply for PCIe clock circuitries: 1.05V
VCCA_XTAL_1P05	Analog supply for XTAL circuitries: 1.05V
VCCDUSB_1P05	Supply for USB digital logic: 1.05V
VCCAPLL_1P05	Analog supply for BCLK/DMI/Audio PLLs: 1.05V. This rail can be derived from the VCCPRIM_1P05 rail with the proper isolation.
VCCPRIM_1P05	Primary Well: 1.05V. For PCIe*/DMI/USB 3.1/SATA MPHY logic, I/O blocks, SRAM, JTAG, CNVi.
VCCDSW_1P05	Deep Sx Well: 1.05V. This rail is generated by on die DSW low dropout (LDO) linear regulator to supply DSW core logic. When the primary well is powered, under PMC control, the regulation in the DSW LDO is turned off and the bypass option is used to derive DSW 1.05V supply from primary well 1.05V supply.
VCCPRIM_MPHY_1P05	Mod PHY Primary: 1.05V. Primary supply for PCIe/USB 3.1/SATA MPHY logic and PCIe/USB PLL dividers.
VCCAMPHYPLL_1P05	Analog supply for USB 3.1, PCIe Gen 2/Gen 3, and SATA 6 Gb/s PLLs: 1.05V.
VCCPRIM_1P8	1.8V Primary Well.
VCCPRIM_3P3	3.3V Primary Well.
VCCSPI	SPI Primary Well 3.3V or 1.8V, for SPI interface.
VCCHDA	3.3V, 1.8V, or 1.5V Well for Intel® High Definition Audio.
VCCDSW_3P3	3.3V Deep Sx Well.
VCCRTC	RTC Well Supply. This rail can drop to 2.0V 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 3.0V. 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.
DCPRTC	RTC decoupling capacitor only. This rail should NOT be driven.
VCCDPHY_1P24	1.24V for CNVi logic. This rail is generated internally with a LDO and needs to be routed to the motherboard so that the rail can be supplied back to the SoC.
VCCPHVLDO_1P8	1.8V Primary Well. On the motherboard, this power pin must be connected to VCCPRIM_1P8 rail in Internal 1.8 V VRM Mode and left as no-connect in External 1.8V VRM Mode.
VCCGPPA	1.8V or 3.3V for GPP_A group.


Table 8-1. Power Rails Descriptions PCH-H (Sheet 2 of 2)

Name	Description
VCCPGPPBC	1.8V or 3.3V for GPP_B and GPP_C groups.
VCCPGPPD	1.8V or 3.3V for GPP_D group.
VCCPGPPEF	1.8V or 3.3V for GPP_E and GPP_F groups.
VCCPGPPG_3P3	3.3V for GPP_G group.
VCCPGPPHK	1.8V or 3.3V for GPP_H and GPP_K groups.
VCCMPHY_SENSE	1.05V Sense Line. This power pin should be left as no-connect.
VSSMPHY_SENSE	0V (Ground) Sense Line. This power pin should be left as no-connect.
VSS	Ground.

§ §

9 Pin Straps

The following signals are used for static configuration. They are sampled at the rising edge of DSW_PWROK, RSMRST#, or 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.

The PCH implements soft straps, which are used to configure specific functions within the PCH and processor very early in the boot process before BIOS or software intervention. The PCH will read soft strap data out of the SPI device prior to the de-assertion of reset to both the Intel® Management Engine and the Host system.

Table 9-1. Pin Straps (Sheet 1 of 4)

Signal	Usage	When Sampled	Comment
GPP_B14 / SPKR	Top Swap Override	Rising edge of PCH_PWROK	<p>The signal has a weak 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 (FWH), 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.</p> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after PCH_PWROK is high. Software will not be able to clear the Top Swap bit until the system is rebooted. The status of this strap is readable using the Top Swap bit (Bus 0, Device 31, Function 0, offset DCh, bit4). This signal is in the primary well.
GPP_B18 / GSPIO_MOSI	No Reboot	Rising edge of PCH_PWROK	<p>The signal has a weak 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.</p> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after PCH_PWROK is high. This signal is in the primary well.
GPP_C2 / SMBALERT#	TLS Confidentiality	Rising edge of RSMRST#	<p>This signal has a weak internal pull-down. 0 = Disable Intel® CSME Crypto Transport Layer Security (TLS) cipher suite (no confidentiality). (Default) 1 = Enable Intel® CSME Crypto Transport Layer Security (TLS) cipher suite (with confidentiality). Must be pulled up to support Intel® AMT with TLS.</p> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after RSMRST# de-asserts. This signal is in the primary well.



Table 9-1. Pin Straps (Sheet 2 of 4)

Signal	Usage	When Sampled	Comment						
GPP_B22 / GSPi1_MOSI	Boot BIOS Strap Bit BBS	Rising edge of PCH_PWROK	<p>This Signal has a weak internal pull-down.</p> <p>This field determines the destination of accesses to the BIOS memory range. Also controllable using Boot BIOS Destination bit (Bus0, Device31, Function0, offset DCh, bit 6).</p> <table border="1"> <thead> <tr> <th>Bit 6</th> <th>Boot BIOS Destination</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>SPI (Default)</td> </tr> <tr> <td>1</td> <td>LPC</td> </tr> </tbody> </table> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after PCH_PWROK is high. If option 1 (LPC) is selected, BIOS may still be placed on LPC, but all platforms are required to have SPI flash connected directly to the PCH's SPI bus with a valid descriptor in order to boot. Boot BIOS Destination select to LPC by functional strap or using Boot BIOS Destination bit will not affect SPI accesses initiated by Intel® CSME or Integrated GbE LAN. This signal is in the primary well. 	Bit 6	Boot BIOS Destination	0	SPI (Default)	1	LPC
Bit 6	Boot BIOS Destination								
0	SPI (Default)								
1	LPC								
GPP_C5 / SML0ALERT#	eSPI or LPC Select	Rising edge of RSMRST#	<p>This signal has a weak internal pull-down.</p> <p>0 = LPC is selected (for EC). (Default)</p> <p>1 = eSPI is selected (for EC).</p> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after RSMRST# de-asserts. This signal is in the primary well. <p>Warning: If this strap is configured to '0' (eSPI is disabled), the eSPI Flash Sharing Mode strap must be configured to '0' as well (SAFS is disabled)</p>						
SPI0_MOSI	Reserved	Rising edge of RSMRST#	<p>External pull-up is required. Recommend 100 kohm if pulled up to 3.3V or 75 kohm if pulled up to 1.8V.</p> <p>This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.</p>						
GPP_H15 / SML3ALERT#	Reserved	Rising edge of RSMRST#	<p>External pull-up is required. Recommend 100K if pulled up to 3.3V or 75K if pulled up to 1.8V.</p> <p>This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.</p>						
GPP_B23 / SML1ALERT# / PCHHOT#	Intel® DCI-OOB	Rising edge of RSMRST#	<p>This signal has an internal pull-down.</p> <p>0 = Disable Intel® DCI-OOB (Default)</p> <p>1 = Enable Intel® DCI-OOB</p> <p>Notes:</p> <ol style="list-style-type: none"> The internal pull-down is disabled after RSMRST# de-asserts. When used as PCHHOT# and strap low, a 150 kohm pull-up is needed to ensure it does not override the internal pull-down strap sampling. <p>This signal is in the primary well.</p>						
SPI0_IO2	Reserved	Rising edge of RSMRST#	<p>External pull-up is required. Recommend 100 kohm if pulled up to 3.3V or 75 kohm if pulled up to 1.8V.</p> <p>This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.</p>						



Table 9-1. Pin Straps (Sheet 3 of 4)

Signal	Usage	When Sampled	Comment
SPIO_IO3	Reserved	Rising edge of RSMRST#	External pull-up is required. Recommend 100 kohm if pulled up to 3.3V or 75 kohm if pulled up to 1.8V. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.
HDA_SDO / I2S0_TXD	Flash Descriptor Security Override	Rising edge of PCH_PWROK	This signal has a weak internal pull-down. 0 = Enable security measures defined in the Flash Descriptor. (Default) 1 = Disable Flash Descriptor Security (<u>override</u>). This strap should 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.
GPP_H12 / SML2ALERT#	eSPI Flash Sharing Mode	Rising edge of RSMRST#	This signal has a weak internal pull-down. 0 = Master Attached Flash Sharing (MAFS) enabled (Default) 1 = Slave Attached Flash Sharing (SAFS) enabled. Notes: 1. The internal pull-down is disabled after RSMRST# de-asserts. 2. This signal is in the primary well. Warning: This strap must be configured to '0' (SAFS is disabled) if the eSPI or LPC strap is configured to '0' (eSPI is disabled)
GPP_I6 / DDPB_C-TRLDATA	Display Port B Detected	Rising edge of PCH_PWROK	This signal has a weak internal pull-down. 0 = Port B is not detected. (Default) 1 = Port B is detected. Notes: 1. The internal pull-down is disabled after PCH_PWROK de-asserts. 2. This signal is in the primary well.
GPP_I8 / DDPC_C-TRLDATA	Display Port C Detected	Rising edge of PCH_PWROK	This signal has a weak internal Pull-down. 0 = Port C is not detected. (Default) 1 = Port C is detected. Notes: 1. The internal pull-down is disabled after PCH_PWROK de-asserts. 2. This signal is in the primary well.
GPP_I10 / DDPD_CTRLDATA	Display Port D Detected	Rising edge of PCH_PWROK	This signal has a weak internal pull-down. 0 = Port D is not detected. (Default) 1 = Port D is detected. Notes: 1. The internal pull-down is disabled after PCH_PWROK de-asserts. 2. This signal is in the primary well.
ITP_PMODE	Reserved	Rising edge of RSMRST#	This signal has a weak internal pull-up. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.



Table 9-1. Pin Straps (Sheet 4 of 4)

Signal	Usage	When Sampled	Comment
GPP_F23 / DDPF_CTRLDATA	Display Port F Detected	Rising edge of PCH_PWROK	This signal has a weak internal pull-down. 0 = Port F is not detected. (Default) 1 = Port F is detected. Notes: 1. The internal pull-down is disabled after PCH_PWROK de-asserts. 2. This signal is in the primary well. 3. This strap applies to platforms that support Display Port F only. Refer to the platform's processor documentation for info on Display Port F support.
GPP_J4 / CNV_BRI_DT / UART0B_RTS#	XTAL Frequency Select	Rising edge of RSMRST#	This signal has a weak internal pull-down. An external pull-up is required on this strap since 38.4 MHz XTAL is not supported on the PCH. 0 = 38.4 XTAL frequency selected. (Default; Not supported) 1 = 24MHz XTAL frequency selected. Notes: 1. The internal pull-down is disabled after RSMRST# de-asserts. 2. This signal is in the primary well.
GPP_J6 / CNV_RGI_DT / UART0B_TXD	M.2 CNV Mode Select	Rising edge of RSMRST#	A weak external pull-up is required. 0 = Integrated CNVi enable. 1 = Integrated CNVi disable. Note: When a RF companion chip is connected to the PCH CNVi interface, the device internal pull-down resistor will pull the strap load to enable CNVi interface.
GPP_J9 / CNV_ MFUART2_TXD	1.8V VCCPSPI	Rising edge of RSMRST#	The signal has a weak internal pull-down 0 = VCCSPI is connected to 3.3V rail 1 = VCCSPI is connected to 1.8V rail Note: If VCCSPI is connected to 1.8V rail, this pin strap must be a '1' for the proper functionality of the SPI (Flash) I/Os
GPD7	Reserved	Rising edge of DSW_PWROK	External pull-up is required. Recommend 100 kohm. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling

§ §



10 Electrical and Thermal Characteristics

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

10.1 Power Rail Absolute Maximum Ratings

Table 10-1. PCH Absolute Power Rail Maximum and Minimum Ratings

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

Note: Overshoot and undershoot voltage guidelines for I/O signals are outlined in [Section 10.5](#).

Table 10-1 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.

10.2 Thermal Specification

10.3 General DC Characteristics

Table 10-2. PCH I_{cc}^3 with Integrated 1.8V VRM Mode OFF (Desktop SKUs) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 I_{ccmax} Current ² (A)	Sx I_{cc} Idle Current ⁴ (mA)	S0ix I_{ccmax} Current (mA)	Deep Sx I_{cc} Idle Current (mA)	G3 (μ A)
VCCAPLL_1P05	1.05	0.041	0.2	0.801	0	0
VCCA_BCLK_1P05	1.05	0.009	0.1	0.321	0	0
VCCA_SRC_1P05	1.05	0.169	0.313	3.092	0	0
VCCA_XTAL_1P05	1.05	0.005	0.551	0.195	0	0

Table 10-2. PCH I_{cc}³ with Integrated 1.8V VRM Mode OFF (Desktop SKUs) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (μA)
VCCAMPHYLL_1P05	1.05	0.145	0.4	4.344	0	0
VCCPRIM_1P05	1.05	7.169+ HSIO Lane Adder refer to Table 10-6 column DT	121.697	1.622 A	0	0
VCCPRIM_MPHY_1P05	1.05	0.106	0.307	4.313	0	0
VCCDSW_1P05	1.05	0.01	0.2	0.003	0.2	0
VCCDUSB_1P05	1.05	0.421	5.698	60.397	0	0
VCCHDA	3.3	0.008	0.1	4.998	0	0
	1.8	0.005	0.06	3	0	0
	1.5	0.004	0.05	2.5	0	0
VCCDSW_3P3	3.3	0.113	0.534	0.705	1.05	0
VCCPRIM_3P3	3.3	0.383	0.391	3.378	0	0
VCCPGPPA	3.3	0.102	0.135	0.378	0	0
	1.8	0.061	0.121	0.340	0	0
VCCPGPPBC	3.3	0.344	0.262	0.853	0	0
	1.8	0.206	0.236	0.767	0	0
VCCPGPPD	3.3	0.14	0.122	0.402	0	0
	1.8	0.084	0.110	0.362	0	0
VCCPGPPEF	3.3	0.175	0.2	0.345	0	0
	1.8	0.104	0.180	0.310	0	0
VCCPGPPG_3P3	3.3	0.145	0.1	0.072	0	0
VCCPGPPHK	3.3	0.263	0.2	0.509	0	0
	1.8	0.157	0.180	0.458	0	0
VCCPRIM_1P8	1.8	0.183+ CNVi Adder refer to Table 10-7, column DT	6.835	18.202	0	0
VCCRTC ¹	3.0	0.60 mA	0.317	0.075	0.316	6
VCCSPI	3.3	0.05	0.1	0.153	0	0
	1.8	0.03	0.09	0.138	0	0
Notes:						
1. The VCC rail ICC data is taken at 3.0 V while the system is in a mechanical off (G3) state at room temperature.						
2. Iccmax estimates assumes 110 °C.						
3. The Iccmax value is a steady state current that can happen after respective power ok has asserted (or reset signal has de-asserted).						
4. Sx Icc Idle assumes PCH is idle and Intel® CSME is power gated.						



Table 10-3. PCH I_{cc}³ with Integrated 1.8V VRM Mode OFF (H Mobile SKUs) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (µA)
VCCAPLL_1P05	1.05	0.034	0.2	0.801	0	0
VCCA_BCLK_1P05	1.05	0.007	0.1	0.087	0	0
VCCA_SRC_1P05	1.05	0.141	0.3	0.838	0	0
VCCA_XTAL_1P05	1.05	0.005	0.544	0.195	0	0
VCCAMPHYPLL_1P05	1.05	0.114	0.4	1.192	0	0
VCCPRIM_1P05	1.05	4.174+ HSIO Lane Adder refer to Table 10-6 column HALO	40.344	0.477 A	0	0
VCCPRIM_MPHY_1P05	1.05	0.088	0.2	1.22	0	0
VCCDSW_1P05	1.05	0.01	0.2	0.001	0.2	0
VCCDUSB_1P05	1.05	0.33	1.288	16.373	0	0
VCCHDA	3.3	0.006	0.1	4.908	0	0
	1.8	0.004	0.06	2.944	0	0
	1.5	0.003	0.05	2.454	0	0
VCCDSW_3P3	3.3	0.094	0.2	0.705	1.05	0
VCCPRIM_3P3	3.3	0.318	0.3	0.916	0	0
VCCPGPPA	3.3	0.085	0.1	0.103	0	0
	1.8	0.051	0.09	0.092	0	0
VCCPGPPBC	3.3	0.286	0.2	0.232	0	0
	1.8	0.171	0.18	0.208	0	0
VCCPGPPD	3.3	0.117	0.1	0.109	0	0
	1.8	0.070	0.09	0.098	0	0
VCCGPPEF	3.3	0.145	0.2	0.094	0	0
	1.8	0.087	0.18	0.084	0	0
VCCPGPPG_3P3	3.3	0.121	0.1	0.072	0	0
VCCGPPHK	3.3	0.219	0.2	0.138	0	0
	1.8	0.131	0.18	0.124	0	0
VCCPRIM_1P8	1.8	0.152+ CNVi Adder refer to Table 10-7, column HALO	6.607	9.411	0	0
VCCRTC ¹	3.0	0.31 mA	0.299	0.075	0.316	6
VCCSPI	3.3	0.042	0.1	0.153	0	0
	1.8	0.025	0.09	0.138	0	0



Table 10-3. PCH I_{CC}³ with Integrated 1.8V VRM Mode OFF (H Mobile SKUs) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (µA)
Notes: 1. The VCC rail ICC data is taken at 3.0 V while the system is in a mechanical off (G3) state at room temperature. 2. Iccmax estimates assumes 110 °C. 3. The Iccmax value is a steady state current that can happen after respective power ok has asserted (or reset signal has de-asserted). 4. Sx Icc Idle assumes PCH is idle and Intel® CSME is power gated.						

Table 10-4. PCH I_{CC}³ with Integrated 1.8V VRM Mode ON (Desktop SKUs) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (µA)
VCCAPLL_1P05	1.05	0.041	0.2	0.801	0	0
VCCA_BCLK_1P05	1.05	0.009	0.1	0.321	0	0
VCCA_SRC_1P05	1.05	0.169	0.313	3.092	0	0
VCCA_XTAL_1P05	1.05	0.005	0.551	0.195	0	0
VCCAMPHYPLL_1P05	1.05	0.145	0.4	4.344	0	0
VCCPRIM_1P05	1.05	7.169+ HSIO Lane Adder refer to Table 10-6, column DT	121.697	1.622 A	0	0
VCCPRIM_MPHY_1P05	1.05	0.106	0.307	4.313	0	0
VCCDSW_1P05	1.05	0.01	0.2	0.003	0.2	0
VCCDUSB_1P05	1.05	0.421	5.698	60.397	0	0
VCCHDA	3.3	0.008	0.1	4.998	0	0
	1.8	0.005	0.06	3	0	0
	1.5	0.004	0.05	2.5	0	0
VCCDSW_3P3	3.3	0.113	0.534	0.705	1.05	0
VCCPRIM_3P3	3.3	0.761+ CNVi Adder refer to Table 10-8, column DT	9.451	30.811	0	0
VCCPGPPA	3.3	0.102	0.135	0.378	0	0
	1.8	0.061	0.121	0.340	0	0
VCCGPPBC	3.3	0.344	0.262	0.853	0	0
	1.8	0.206	0.236	0.767	0	0
VCCGPPD	3.3	0.14	0.122	0.402	0	0
	1.8	0.084	0.110	0.362	0	0
VCCGPPEF	3.3	0.175	0.2	0.345	0	0
	1.8	0.104	0.18	0.310	0	0
VCCGPPG_3P3	3.3	0.145	0.1	0.072	0	0



Table 10-4. PCH I_{cc}³ with Integrated 1.8V VRM Mode ON (Desktop SKUs) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (µA)
VCCPGPPHK	3.3	0.263	0.2	0.509	0	0
	1.8	0.157	0.18	0.458	0	0
VCCRTC	3.0	0.60 mA	0.317	0.075	0.316	6
VCCSPI	3.3	0.05	0.1	0.153	0	0
	1.8	0.03	0.09	0.138	0	0

Notes:

- The VCC rail ICC data is taken at 3.0 V while the system is in a mechanical off (G3) state at room temperature.
- Iccmax estimates assumes 110 °C.
- The Iccmax value is a steady state current that can happen after respective power ok has asserted (or reset signal has de-asserted).
- Sx Icc Idle assumes PCH is idle and Intel® CSME is power gated.

Table 10-5. PCH I_{cc}³ with Integrated 1.8V VRM Mode ON (H Mobile SKUs) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (µA)
VCCAPLL_1P05	1.05	0.034	0.2	0.801	0	0
VCCA_BCLK_1P05	1.05	0.007	0.1	0.087	0	0
VCCA_SRC_1P05	1.05	0.141	0.3	0.838	0	0
VCCA_XTAL_1P05	1.05	0.005	0.544	0.195	0	0
VCCAMPHYPLL_1P05	1.05	0.114	0.4	1.192	0	0
VCCPRIM_1P05	1.05	4.174+ HSIO Lane Adder refer to Table 10-6, column HALO	40.344	0.477 A	0	0
VCCPRIM_MPHY_1P05	1.05	0.088	0.2	1.22	0	0
VCCDSW_1P05	1.05	0.01	0.2	0.001	0.2	0
VCCDUSB_1P05	1.05	0.33	1.288	16.373	0	0
VCCHDA	3.3	0.007	0.1	4.908	0	0
VCCDSW_3P3	3.3	0.094	0.2	0.705	1.05	0
VCCPRIM_3P3	3.3	0.536+ CNVi Adder refer to Table 10-8, column HALO	8.732	17.105	0	0
VCCPGPPA	3.3	0.085	0.1	0.103	0	0
VCCPGPPBC	3.3	0.286	0.2	0.232	0	0
VCCPGPPD	3.3	0.117	0.1	0.109	0	0
VCCPGPPEF	3.3	0.145	0.2	0.094	0	0
VCCPGPPG_3P3	3.3	0.121	0.1	0.072	0	0
VCCPGPPHK	3.3	0.219	0.2	0.138	0	0
VCCRTC	3.0	0.31 mA	0.299	0.075	0.316	6


Table 10-5. PCH I_{cc}³ with Integrated 1.8V VRM Mode ON (H Mobile SKUs) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current ² (A)	Sx Icc Idle Current ⁴ (mA)	S0ix Iccmax Current (mA)	Deep Sx Icc Idle Current (mA)	G3 (μA)
VCCSPI	3.3	0.042	0.1	0.153	0	0

Notes:

- The VCC rail ICC data is taken at 3.0 V while the system is in a mechanical off (G3) state at room temperature.
- Iccmax estimates assumes 110 °C.
- The Iccmax value is a steady state current that can happen after respective power ok has asserted (or reset signal has de-asserted).
- Sx Icc Idle assumes PCH is idle and Intel® CSME is power gated.
- Sx Icc at 3.3 V level is assumed. Sx Icc data at the 1.8 V and/or 1.5 V level not measured.

Table 10-6. PCH VCCPRIM_1P05 Adder per HSIO Lane

Details	Icc (HALO) (mA)	Icc (DT) (mA)
Baseline (x4 DMI Gen3)	639	1116
Baseline (x4 DMI Gen2)	607	1084
Each PCIe Gen3 Lane	148	178
Each PCIe Gen2 Lane	123	152
Each USB 3.1 Gen 2 (10 Gb/s) Port	182	221
Each USB 3.1 Gen 1 (5 Gb/s) Port	125	155
Integrated GbE Port	62	98
Each SATA 6 Gb/s Port	126	167

Table 10-7. PCH VCCPRIM_1P8 Adder with and without CNVi

Details	Icc (HALO) (mA)	Icc (DT) (mA)
Without CNVi	0	0
With CNVi	582	589

Table 10-8. PCH VCCPRIM_3P3 Adder with and without CNVi

Details	Icc (HALO) (mA)	Icc (DT) (mA)
Without CNVi	0	0
With CNVi	582	589



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 1 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
<p>NOTE: For GPIO pads (GPP) 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 GPIO Chapter for the multiplexed functions on a specific GPIO pad.</p> <p>Associated Signals¹: GPP_A0/RCIN#/ESPI_ALERT1#, GPP_A1/LAD0/ESPI_IO0, GPP_A2/LAD1/ESPI_IO1, GPP_A3/LAD2/ESPI_IO2, GPP_A4/LAD3/ESPI_IO3, GPP_A5/LFRAME#/ESPI_CS0#, GPP_A6/SERIRQ/ESPI_CS1#, GPP_A7/PIRQA#/ESPI_ALERT0#, GPP_A8/CLKRUN#, GPP_A9/CLKOUT_LPC0/ESPI_CLK, GPP_A10/CLKOUT_LPC1, GPP_A11/PME#/SD_VDD2_PWR_EN#, GPP_A12/BM_BUSY#/ISH_GP6/SX_EXIT_HOLDOFF#, GPP_A13/SUSWARN#/SUSPWRDNACK, GPP_A14/SUS_STAT#/ESPI_RESET#, GPP_A15/SUSACK#, GPP_A16/CLKOUT_48, GPP_A17/SD_VDD1_PWR_EN#/ISH_GP7, GPP_A18/ISH_GP0, GPP_A19/ISH_GP1, GPP_A20/ISH_GP2, GPP_A21/ISH_GP3, GPP_A22/ISH_GP4, GPP_A23/ISH_GP5, GPP_D17/DMIC_CLK1/SNDW3_CLK, GPP_D18/DMIC_DATA1/SNDW3_DATA, GPP_D19/DMIC_CLK0/SNDW4_CLK, GPP_D20/DMIC_DATA0/SNDW4_DATA, HDA_BCLK/I2S0_SCLK, HDA_RST#/I2S1_SCLK, HDA_SYNC/I2S0_SFRM, HDA_SDO/I2S0_TXD, HDA_SDI0/I2S0_RXD, HDA_SDI1/I2S1_RXD, I2S1_SFRM / SNDW2_CLK, I2S1_TXD / SNDW2_DATA</p>							
3.3V 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	—	13	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU (WeakPull-Up) Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
	R _{pd}	WPD (Weak Pull-Down) Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
1.8V 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	—	13	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
<p>Notes:</p> <p>1. For GPIO supported voltages, refer to GPIO chapter.</p>							



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 2 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
<p>NOTE: For GPIO pads (GPP) 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 GPIO Chapter for the muxed functions on a specific GPIO pad.</p> <p>Associated Signals¹: GPP_B0/GSPI0_CS1#, GPP_B1/GSPI1_CS1#/TIME_SYNC1, GPP_B2/VRALERT#, GPP_B3/CPU_GP2, GPP_B4/CPU_GP3, GPP_B5/SRCCLKREQ0#, GPP_B6/SRCCLKREQ1#, GPP_B7/SRCCLKREQ2#, GPP_B8/SRCCLKREQ3#, GPP_B9/SRCCLKREQ4#, GPP_B10/SRCCLKREQ5#, GPP_B11/I2S_MCLK, GPP_B12/SLP_S0#, GPP_B13/PLTRST#, GPP_B14/SPKR, GPP_B15/GSPI0_CS0#, GPP_B16/GSPI0_CLK, GPP_B17/GSPI0_MISO, GPP_B18/GSPI0_MOSI, GPP_B19/GSPI1_CS0#, GPP_B20/GSPI1_CLK, GPP_B21/GSPI1_MISO, GPP_B22/GSPI1_MOSI, GPP_B23/SML1ALERT#/PCHHOT#, GPP_C0/SMBCLK, GPP_C1/SMBDATA, GPP_C2/SMBALERT#, GPP_C3/SML0CLK, GPP_C4/SML0DATA, GPP_C5/SML0ALERT#, GPP_C6/SML1CLK, GPP_C7/SML1DATA, GPP_C8/UART0_RXD, GPP_C9/UART0_TXD, GPP_C10/UART0_RTS#, GPP_C11/UART0A_CTS#, GPP_C12/UART1_RXD/ISH_UART1_RXD, GPP_C13/UART1_TXD/ISH_UART1_TXD, GPP_C14/UART1_RTS#/ISH_UART1_RTS#, GPP_C15/UART1_CTS#/ISH_UART1_CTS#, GPP_C16/I2C0_SDA, GPP_C17/I2C0_SCL, GPP_C18/I2C1_SDA, GPP_C19/I2C1_SCL, GPP_C20/UART2_RXD, GPP_C21/UART2_TXD, GPP_C22/UART2_RTS#, GPP_C23/UART2_CTS#, GPP_D0/SPI1_CS#/SBK0/BK0, GPP_D1/SPI1_CLK/SBK1/BK1, GPP_D2/SPI1_MISO/SBK2/BK2, GPP_D3/SPI1_MOSI/SBK3/BK3, GPP_D4/ISH_I2C2_SDA/I2C3_SDA/SBK4/BK4, GPP_D5/I2S2_SFRM/CNV_RF_RESET#, GPP_D6/I2S2_TXD/MODEM_CLKREQ, GPP_D7/I2S2_RXD, GPP_D8/I2S2_SCL, GPP_D9/ISH_SPI_CS#/GSPi2_CS0#, GPP_D10/ISH_SPI_CLK/GSPi2_CLK, GPP_D11/ISH_SPI_MISO/GP_BSSB_CLK/GSPi2_MISO, GPP_D12/ISH_SPI_MOSI/GP_BSSB_DI/GSPi2_MOSI, GPP_D13/ISH_UART0_RXD/I2C2_SDA, GPP_D14/ISH_UART0_TXD/I2C2_SCL, GPP_D15/ISH_UART0_RTS#/GSPi2_CS1#/CNV_WFEN, GPP_D16/ISH_UART0_CTS#/CNV_WCEN, GPP_D21/SPI1_IO2, GPP_D22/SPI1_IO3, GPP_D23/ISH_I2C2_SCL/I2C3_SCL, GPD0/BATLOW#, GPD1/ACPRESENT, GPD2/LAN_WAKE#, GPD3/PWRBTN#, GPD4/SLP_S3#, GPD5/SLP_S4#, GPD6/SLP_A#, GPD7, GPD8/SUSCLK, GPD9/SLP_WLAN#, GPD10/SLP_S5#, GPD11/LANPHYPC, SLP_LAN#, SLP_SUS#, GPD2/LAN_WAKE#, DRAM_RESET#</p>							
3.3V 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	-10	10	µA		
	C _{IN}	Input Pin Capacitance	—	14	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	1K-50% 5K-70% 20K-25%	1K+100% 5K+70% 20K+25%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ω		
1.8V 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	-10	10	µA		
	C _{IN}	Input Pin Capacitance	—	14	pF		



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 3 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45		V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	1K-50% 5K-70% 20K-25%	1K+100% 5K+70% 20K+25%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
Notes:							
1. For GPIO supported voltages, refer to GPIO chapter.							
NOTE: For GPIO pads (GPP) 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 GPIO Chapter for the muxed functions on a specific GPIO pad.							
Associated Signals ¹ : GPP_G0/SD_CMD, GPP_G1/SD_DATA0, GPP_G2/SD_DATA1, GPP_G3/SD_DATA2, GPP_G4/SD_DATA3, GPP_G5/SD_CD#, GPP_G6/SD_CLK, GPP_G7/SD_WP							
3.3V 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	—	13	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
1.8V 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	—	13	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+25%	Ω		



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 4 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Notes:							
1. For GPIO supported voltages, refer to GPIO chapter.							
NOTE: For GPIO pads (GPP) 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 GPIO Chapter for the muxed functions on a specific GPIO pad.							
Associated Signals: GPP_K0, GPP_K1, GPP_K2, GPP_K3, GPP_K4, GPP_K5, GPP_K6, GPP_K7, GPP_K8, GPP_K9, GPP_K10, GPP_K11, GPP_K12/GSXDOUT, GPP_K13/GSXSLOAD, GPP_K14/GSXDIN, GPP_K15/GSXSRESET#, GPP_K16/GSXCLK, GPP_K17/ADR_COMPLETE, GPP_K18/NMI#, GPP_K19/SMI#, GPP_K20, GPP_K21, GPP_K22/IMGCLKOUT0, GPP_K23/IMGCLKOUT1, GPP_H0/SRCCLKREQ6#, GPP_H1/SRCCLKREQ7#, GPP_H2/SRCCLKREQ8#, GPP_H3/SRCCLKREQ9#, GPP_H4/SRCCLKREQ10#, GPP_H5/SRCCLKREQ11#, GPP_H6/SRCCLKREQ12#, GPP_H7/SRCCLKREQ13#, GPP_H8/SRCCLKREQ14#, GPP_H9/SRCCLKREQ15#, GPP_H10/SML2CLK, GPP_H11/SML2DATA, GPP_H12/SML2ALERT#, GPP_H13/SML3CLK, GPP_H14/SML3DATA, GPP_H15/SML3ALERT#, GPP_H16/SML4CLK, GPP_H17/SML4DATA, GPP_H18/SML4ALERT#, GPP_H19/ISH_I2C0_SDA, GPP_H20/ISH_I2C0_SCL, GPP_H21/ISH_I2C1_SDA, GPP_H22/ISH_I2C1_SCL, GPP_H23/TIME_SYNC0, GPP_I0/DDPB_HPDP0/DISP_MISC0, GPP_I1/DDPC_HPDP1/DISP_MISC1, GPP_I2/DDPD_HPDP2/DISP_MISC2, GPP_I3/DDPF_HPDP3/DISP_MISC3, GPP_I4/EDP_HPDP/DISP_MISC4, GPP_I5/DDPB_CTRLCLK, GPP_I6/DDPB_CTRLDATA, GPP_I7/DDPC_CTRLCLK, GPP_I8/DDPC_CTRLDATA, GPP_I9/DDPD_CTRLCLK, GPP_I10/DDPD_CTRLDATA, GPP_I11/M2_SKT2_CFG0, GPP_I12/M2_SKT2_CFG1, GPP_I13/M2_SKT2_CFG2, GPP_I14/M2_SKT2_CFG3, SYS_PWROK, SYS_RESET#							
3.3V 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	-10	10	µA		
	C _{IN}	Input Pin Capacitance	—	14	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	1K-50% 5K-70% 20K-35%	1K+100% 5K+70% 20K+35%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ω		
1.8V 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	-10	10	µA		
	C _{IN}	Input Pin Capacitance	—	14	pF		



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 5 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	1K-50% 5K-70% 20K-35%	1K+100% 5K+70% 20K+35%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-35%	5K+70% 20K+35%	Ω		
<p>NOTE: For GPIO pads (GPP) 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 GPIO Chapter for the muxed functions on a specific GPIO pad.</p> <p>Associated Signals: GPP_E0/SATAXPCEI0/SATAGP0, GPP_E1/SATAXPCEI1/SATAGP1, GPP_E2/SATAXPCEI2/SATAGP2, GPP_E3/CPU_GP0, GPP_E4/SATA_DEVSLP0, GPP_E5/SATA_DEVSLP1, GPP_E6/SATA_DEVSLP2, GPP_E7/CPU_GP1, GPP_E8/SATALED#, GPP_E9/USB2_OC0#, GPP_E10/USB2_OC1#, GPP_E11/USB2_OC2#, GPP_E12/USB2_OC3#, GPP_F0/SATAXPCEI3/SATAGP3, GPP_F1/SATAXPCEI4/SATAGP4, GPP_F2/SATAXPCEI5/SATAGP5, GPP_F3/SATAXPCEI6/SATAGP6, GPP_F4/SATAXPCEI7/SATAGP7, GPP_F5/SATA_DEVSLP3, GPP_F6/SATA_DEVSLP4, GPP_F7/SATA_DEVSLP5, GPP_F8/SATA_DEVSLP6, GPP_F9/SATA_DEVSLP7, GPP_F10/SATA_SCLK, GPP_F11/SATA_SLOAD, GPP_F12/SATA_SDATAOUT1, GPP_F13/SATA_SDATAOUT0, GPP_F14/PS_ON#, GPP_F15/USB2_OC4#, GPP_F16/USB2_OC5#, GPP_F17/USB2_OC6#, GPP_F18/USB2_OC7#, GPP_F19/eDP_VDDEN, GPP_F20/eDP_BKLTEN, GPP_F21/eDP_BKLTCTL, GPP_F22/DDPF_CTRLCLK, GPP_F23/DDPF_CTRLDATA, SPI0_IO2, SPI0_IO3, GPP_B16/GSPI0_CLK</p>							
3.3V 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	—	13	pF		
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ω		
1.8V 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	—	13	pF		



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 6 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	VCC	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ω		
	R _{pd}	WPD Resistance	5K-70% 20K-25%	5K+70% 20K+35%	Ω		
Associated Signals: PM_SYNC, PECEI, CPUPWRGD, THRMTRIP#, PLTRST_CPU#, PM_DOWN, TRIGGER_IN, TRIGGER_OUT, PCH_JTAG_TDO, PCH_JTAGX, PRDY#, PREQ#, CPU_TRST#, PCH_JTAG_TDI, PCH_JTAG_TMS, PCH_JTAG_TCK, ITP_PMODE, HDACPU_SDI, HDACPU_SCLK, HDACPU_SDO							
1.05V Operation							
Input	V _{IH}	Input High Voltage Threshold	JTAG: 0.8 x VCC PECEI: 0.7 x VCC iDISPLAY: 0.65 x VCC	—	V		
	V _{IL}	Input Low Voltage Threshold	—	JTAG: 0.45 x VCC PECEI: 0.3 x VCC iDISPLAY: 0.3 x VCC	V		
	I _{IL}	Input Leakage Current	-10	10	μA		
	C _{IN}	Input Pin Capacitance	—	2	pF		
Output	V _{OH}	Output High Voltage Threshold	0.75 x VCC	VCC	V	I _{OH} =6 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.25 x VCC	V	I _{OL(min)} =-0.5 mA I _{OL(max)} =-1 mA	
	R _{pu}	WPU Resistance	1K-30% 20K-30%	1K+30% 20K+30%	Ω		
	R _{pd}	WPD Resistance	1K-30% 20K-30%	1K+30% 20K+30%	Ω		
Associated Signals: GPP_J0/CNV_PA_BLANKING, GPP_J1/CPU_C10_GATE#, GPP_J2, GPP_J3, GPP_J4/CNV_BRI_DT/UART0B_RTS#, GPP_J5/CNV_BRI_RSP/UART0B_RXD, GPP_J6/CNV_RGI_DT/UART0B_TXD, GPP_J7/CNV_RGI_RSP/UART0B_CTS#, GPP_J8/CNV_MFUART2_RXD, GPP_J9/CNV_MFUART2_TXD, GPP_J10, GPP_J11/A4WP_PRESENT							
3.3V Operation							
Input	V _{IH}	Input High Voltage Threshold	0.65 x VCC	—	V		
	V _{IL}	Input Low Voltage Threshold	—	0.35 x VCC	V		
	I _{IL}	Input Leakage Current	-5	5	μA		
	C _{IN}	Input Pin Capacitance	—	5	pF		



Table 10-9. Single-Ended Signal DC Characteristics as Inputs or Outputs (Sheet 7 of 7)

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Output	V _{OH}	Output High Voltage Threshold	VCC - 0.45	—	V	I _{OH} =3 mA	
	V _{OL}	Output Low Voltage Threshold	—	0.45	V	I _{OL} =-3 mA	
	R _{pu}	WPU Resistance	20K-40%	20K+40%	Ω		
	R _{pd}	WPD Resistance	20K-40%	20K+40%	Ω		
1.8V Operation							
Input	V _{IH}	Input High Voltage Threshold	0.65 x VCC	—	V		
	V _{IL}	Input Low Voltage Threshold	—	0.35 x VCC	V		
	I _{IL}	Input Leakage Current	-5	5	μA		
	C _{IN}	Input Pin Capacitance	—	5	pF		
Output	V _{OL}	Output Low Voltage Threshold	VCC - 0.45	—	V	I _{OL} =-3 mA	
			—	0.45			
	R _{pu}	WPU Resistance Threshold	20K-40%	20K+40%	Ω		
	R _{pd}	WPD Resistance	20K-40%	20K+40%	Ω		
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.							

Table 10-10. Single-Ended Signal DC Characteristics as Inputs or Outputs

Type	Symbol	Parameter	Minimum	Maximum	Unit	Condition	Notes
Associated Signals: INTRUDER#, RSMRST#, PCH_PWROK, DSW_PWROK, SRTCST#							
Input	V _{IH}	Input High Voltage Threshold	0.65 x VCCRTC	VCCRTC+0.5	V		1, 2
	V _{IL}	Input Low Voltage Threshold	-0.5	0.3 x VCCRTC	V		2
Associated Signals: RTCRST#							
Input	V _{IH}	Input High Voltage Threshold	0.75 x VCCRTC	VCCRTC+0.5	V		1, 2
	V _{IL}	Input Low Voltage Threshold	-0.5	0.4 x VCCRTC	V		2
Associated Signals: RTCX1#							
Input	V _{IH}	Input High Voltage Threshold	0.8	1.2	V		
	V _{IL}	Input Low Voltage Threshold	-0.5	0.1	V		
Notes:							
1. 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.							
2. These buffers have input hysteresis. V _{IH} levels are for rising edge transitions and V _{IL} levels are for falling edge transitions.							



Table 10-11. Differential Signals Characteristics (Sheet 1 of 3)

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		
VTX_CM-Acp-p	TX AC Common Mode Output Voltage (5GT/s)	—	100	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		
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		
VTX_CM-Acp-p	TX AC Common Mode Output Voltage (8GT/s)	—	100	mV		
ZTX-DIFF-DC	DC Differential TX Impedance	80	120	Ohm		
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	Refer to Stressed Voltage Eye Parameters Table in PCIe GEN3 industry specifications.				
VRX_CM-ACp	AC peak Common Mode Input Voltage	—	150	mV		
Associated Signals: SATA						
VIMIN-Gen1i	Minimum Input Voltage - 1.5Gb/s internal SATA	325	—	mVdiff p-p		2
VIMAX-Gen1i	Maximum Input Voltage - 1.5Gb/s internal SATA	—	600	mVdiff p-p		2
VIMIN-Gen1m	Minimum Input Voltage - 1.5Gb/s eSATA	240	—	mVdiff p-p		2
VIMAX-Gen1m	Maximum Input Voltage - 1.5Gb/s eSATA	—	600	mVdiff p-p		2
VIMIN-Gen2i	Minimum Input Voltage - 3.0Gb/s internal SATA	275	—	mVdiff p-p		2
VIMAX-Gen2i	Maximum Input Voltage - 3.0Gb/s internal SATA	—	750	mVdiff p-p		2
VIMIN-Gen2m	Minimum Input Voltage - 3.0Gb/s eSATA	240	—	mVdiff p-p		2



Table 10-11. Differential Signals Characteristics (Sheet 2 of 3)

Symbol	Parameter	Minimum	Maximum	Unit	Conditions	Notes
VIMAX-Gen2m	Maximum Input Voltage - 3.0Gb/s eSATA	—	750	mVdiff p-p		2
VIMIN-Gen3i	Minimum Input Voltage - 6.0Gb/s internal SATA	240	—	mVdiff p-p		2
VIMAX-Gen3i	Maximum Input Voltage - 6.0Gb/s internal SATA	—	1000	mVdiff p-p		2
VOMIN-Gen1i,m	Minimum Output Voltage 1.5Gb/s internal and eSATA	400	—	mVdiff p-p		3
VOMAX-Gen1i,m	Maximum Output Voltage 1.5Gb/s internal and eSATA	—	600	mVdiff p-p		3
VOMIN-Gen2i,m	Minimum Output Voltage 3.0Gb/s internal and eSATA	400	—	mVdiff p-p		3
VOMAX-Gen2i,m	Maximum Output Voltage 3.0Gb/s internal and eSATA	—	700	mVdiff p-p		3
VOMIN-Gen3i	Minimum Output Voltage 6.0Gb/s internal SATA	200	—	mVdiff p-p		3
VOMAX-Gen3i	Maximum Output Voltage 6.0Gb/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
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
Associated Signals: USB 3.1						
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		



Table 10-11. Differential Signals Characteristics (Sheet 3 of 3)

Symbol	Parameter	Minimum	Maximum	Unit	Conditions	Notes
Associated Signals: CLKOUT_CPUBCLK_P/N, CLKOUT_CPUPCIBCLK_P/N, CLKOUT_ITPXDP_P/N, CLKOUT_PCIE_P/N[15:0]						
Output	V _{Swing}	Differential Output Swing	300	—	mV	12
	V _{Cross}	Crossing Point Voltage	250	550	mV	11,13,14
	V _{Cross_Delta}	Variation of VCROSS	—	140	mV	11,13,17
	V _{Max}	Max Output Voltage	—	1.15	V	11,15
	V _{Min}	Min Output Voltage	-0.3	—	V	11,16
Notes: 1. PCI Express mVdiff p-p = 2* PCIE[x]_TXP - PCIE[x]_TXN ; PCI Express mVdiff p-p = 2* PCIE[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} = USBp[x]P - USBp[x]N 5. Includes VDI range 6. Applies to Low-Speed/Full-Speed USB 2.0 7. Applies to High-Speed USB 2.0. 8. USB 3.1 mVdiff p-p = 2* USB3Rp[x] - USB3Rn[x] ; USB 3.1 mVdiff p-p = 2* USB3Tp[x] - USB3Tn[x] 9. For PCIe, Gen 1, Gen 2 and Gen 3 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. V _{Cross} 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 max. voltage including overshoot 16. The min. voltage including undershoot 17. The total variation of all V _{Cross} measurements in any particular system. Note that this is a subset of V _{Cross} MIN/MAX (V _{Cross} absolute) allowed. The intent is to limit V _{Cross} induced modulation by setting V _{Cross_Delta} to be smaller than V _{Cross} absolute.						

Table 10-12. Power Rail Operational Voltage Range (Sheet 1 of 2)

Symbol	Parameter	Minimum	Nominal	Maximum	Unit	Notes
VCCPRIM_1p05	Core Logic, SRAM, I/O Blocks, USB AFE, Processor Sideband, JTAG, Thermal Sensor Primary WellSP, Analog Supply for Audio PLL Primary Well, Mod-PHY Supply Primary Well	0.9975	1.05	1.1025	V	1
VCCPRIM_3p3	Primary Well for HVCMOS and display, USB2, FUSE, Integrated 1.8V VRM input supply	3.135	3.3	3.465	V	1
VCCPRIM_1P8	Primary Well for 1.8V GPIO and biasing	1.71	1.8	1.89	V	1
VCCPRIM_MPHY_1P05	Primary supply for PCIe/USB PLL dividers.	0.9975	1.05	1.1025	V	1
VCCAMPHYPLL_1p05	Analog Supply for USB 3.0, PCIe Gen2, SATA and PCIe* Gen 3 PLL Primary Well	0.9975	1.05	1.1025	V	1
VCCA_BCLK_1P05	Analog supply for BCLK buffer	0.9975	1.05	1.1025	V	1
VCCA_SRC_1P05	Analog supply for PCIe clock buffer	0.9975	1.05	1.1025	V	1
VCCA_XTAL_1P05	Analog supply for XTAL circuitries	0.9975	1.05	1.1025	V	1
VCCAPLL_1P05	Analog supply for BCLK PLLs	0.9975	1.05	1.1025	V	1
VCCDPHY_1P24	1.24V for CNVi logic (Internal generated)	1.178	1.24	1.302	V	1
VCCDSW_1P05	Deep Sx Well (Internal generated)	0.9975	1.05	1.1025	V	1
VCCDUSB_1P05	Supply for USB AFE	0.9975	1.05	1.1025	V	1



Table 10-12. Power Rail Operational Voltage Range (Sheet 2 of 2)

Symbol	Parameter	Minimum	Nominal	Maximum	Unit	Notes
VCCPHVLD0_1P8	Integrated 1.8V VRM output	1.71	1.8	1.89	V	1
VCCPGPPA	Group A Primary Well GPIOs	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCPGPPBC	Group B and C Primary Well GPIOs	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCPGPPD	Group D Primary Well GPIOs	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCPGPPEF	Group E and F Primary Well GPIOs	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCPGPPG_3P3	Group G Primary Well GPIOs	3.135	3.3	3.465	V	1
VCCPGPPHK	Group H and K Primary Well GPIOs	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCSPI	SPI Primary Well	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
VCCHDA	Intel® HD Audio Supply Primary Well	3.135	3.3	3.465	V	1
		1.71	1.8	1.89	V	1
		1.425	1.5	1.575	V	1
VCCDSW_3p3	Deep Sx Well for GPD and USB 2.0	3.135	3.3	3.465	V	1
VCCRTC	RTC Well Supply	2	3.3	3.465	V	1,2

Notes:

- The I/O buffer supply voltage is measured at the PCH package pins. The tolerances shown in Table 10-12 are inclusive of all noise from DC up to 1 MHz. In testing, the voltage rails should be measured with a bandwidth limited oscilloscope that has a roll off of 3db/decade above 20 MHz.
- Maximum Crystal ESR is 50 kohm.

10.4 AC Characteristics

Table 10-13. PCI Express* Interface Timings (Sheet 1 of 2)

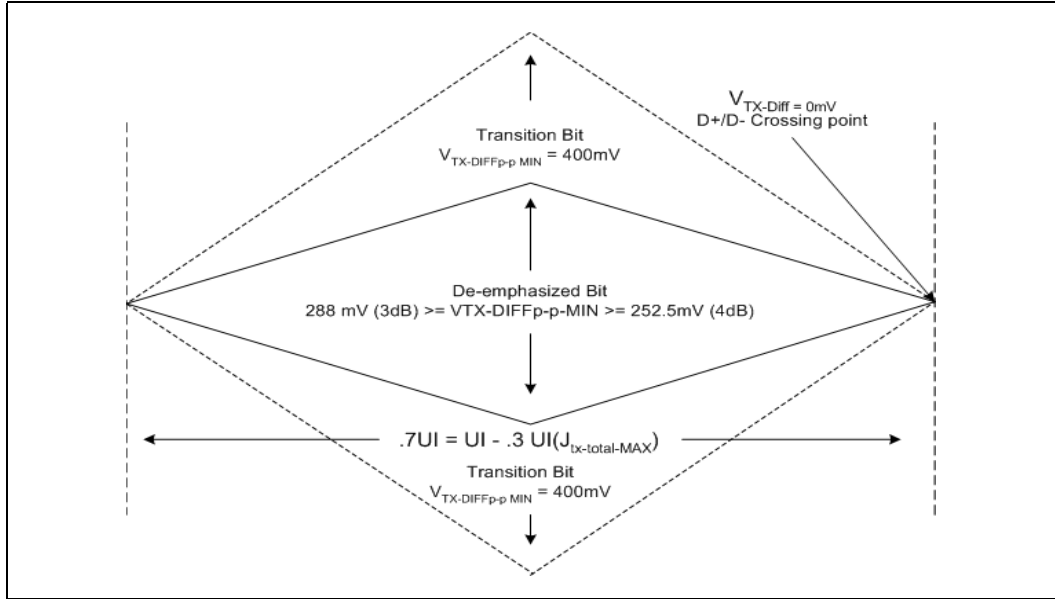
Symbol	Parameter	Minimum	Maximum	Unit	Figures	Notes
Transmitter and Receiver Timings						
UI (Gen1)	Unit Interval – PCI Express*	399.88	400.12	ps		5
UI (Gen 2)	Unit Interval – PCI Express*	199.9	200.1	ps		5
UI (Gen 3)	Unit Interval – PCI Express*	124.96	125.03	ps		
T _{TX-EYE} (Gen 1/ Gen 2)	Minimum Transmission Eye Width	0.75	–	UI	10-1	1,2



Table 10-13. PCI Express* Interface Timings (Sheet 2 of 2)

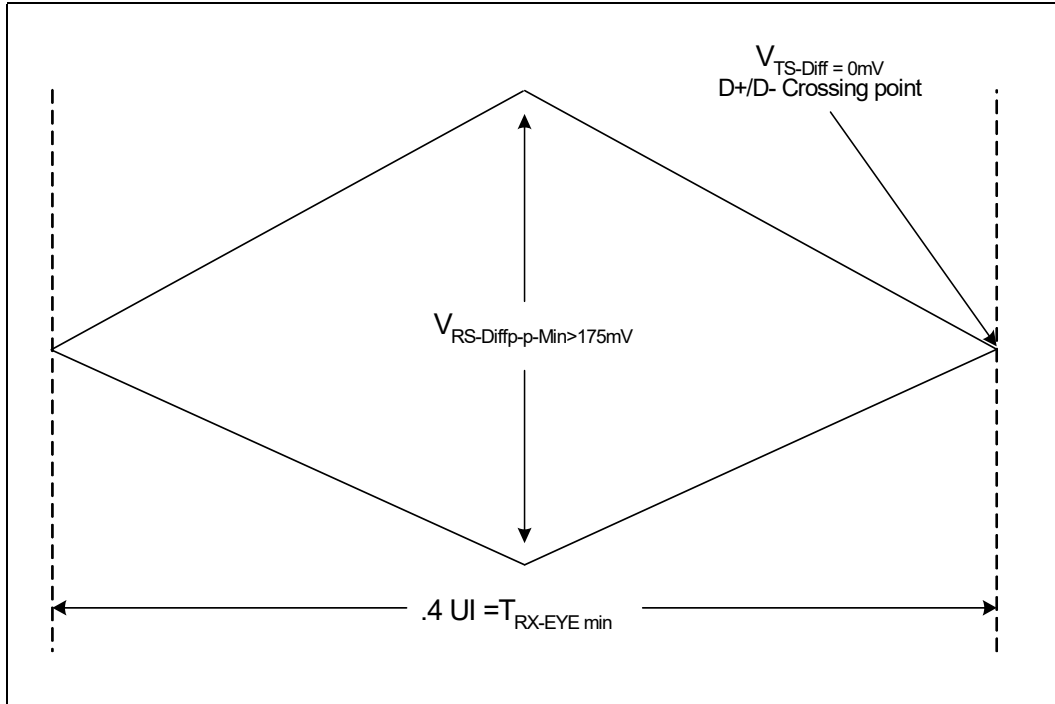
Symbol	Parameter	Minimum	Maximum	Unit	Figures	Notes
Transmitter and Receiver Timings						
$T_{TX-EYE-MEDIAN-to-MAX-JITTER}$ (Gen 1)	Maximum time between the jitter median and maximum deviation from the median	0.125	—	UI		1,2
$T_{TX-EYE-MEDIAN-to-MAX-JITTER}$ (Gen 2)	Maximum time between the jitter median and maximum deviation from the median	0.15	—	UI		
$T_{TX-EYE-MEDIAN-to-MAX-JITTER}$ (Gen 3)	Maximum time between the jitter median and maximum deviation from the median	0.15	—	UI		
T_{RX-EYE} (Gen 1)	Minimum Receiver Eye Width	0.40	—	UI	10-2	3,4
T_{RX-EYE} (Gen 2)	Minimum Receiver Eye Width	0.60	—	UI	10-2	3,4
$T_{Min-Pulse}$ (Gen 2)	Instantaneous Pulse Width	0.9	—	UI		
<p>Notes: Refer to www.pcisig.com for the updated specifications.</p> <ol style="list-style-type: none"> 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) A $T_{TX-EYE} = 0.70$ UI provides for a total sum of deterministic and random jitter budget of $T_{TXJITTER-MAX} = 0.30$ UI for the Transmitter collected over any 250 consecutive TX UIs. The $T_{TXEYE-MEDIAN-to-MAX-JITTER}$ 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. 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. A $T_{RX-EYE} = 0.40$ 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 $T_{RX-EYE-MEDIAN-to-MAX-JITTER}$ 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. Nominal Unit Interval is 400 ps for 2.5 GT/s and 200 ps for 5 GT/s. 						

Figure 10-1. PCI Express* Transmitter Eye



Note: Gen 1 example is shown for the illustration. Refer to www.pcisig.com for the updated specifications.

Figure 10-2. PCI Express* Receiver Eye



Note: Gen 1 example is shown for the illustration. Refer to www.pcisig.com for the updated specifications.



Table 10-14. DDC Characteristics

Signal Group: eDP_VDDEN, eDP_BKLTEN, eDP_BKLTCTL, DDP[F,D:B]_CTRLCLK, DDP[F,D:B]_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.

10.4.1 Panel Power Sequencing and Backlight Control

The PCH integrates 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 10-3. Panel Power Sequencing

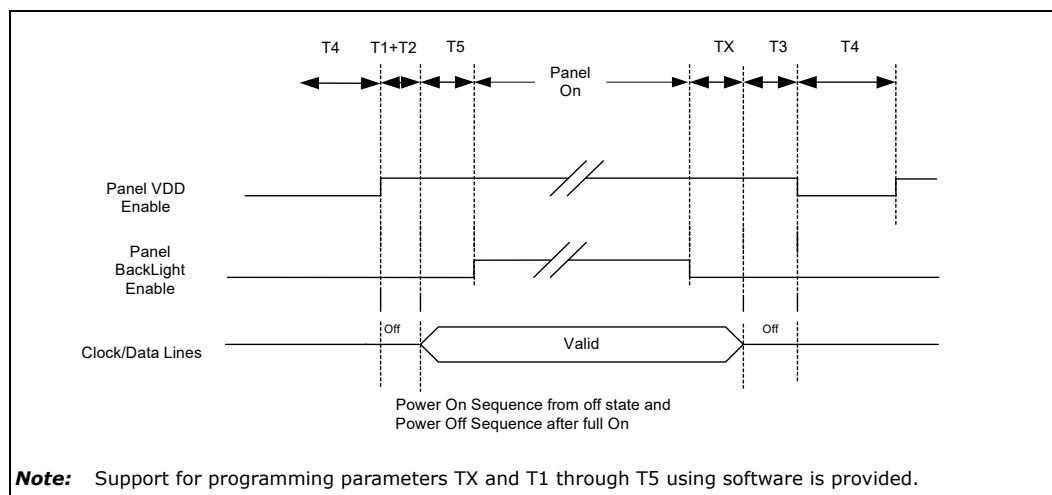




Table 10-15. DisplayPort* Hot-Plug Detect Interface

Signal Group: DDPB_HPD0, DDPC_HPD1, DDPD_HPD2, DDPF_HPD3, eDP_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 10-16. Clock Timings (Sheet 1 of 4)

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
LPC Clock (CLKOUT_LPC[1:0])						
	Period	41.16	42.18	ns		10-4
	High Time	16.67	25.00	ns		10-4
	Low Time	16.67	25.00	ns		10-4
	Duty Cycle	40	60	%		
	Jitter	—	500	ps	8, 9	
	Flight Time (PCH to Device)	—	3	ns		
CLKOUT_PCIE_P/N[15:0], CLKOUT_ITPXD_P/[P,N]						
Clock Period	Period SSC On	9.849	10.201	ns		10-5
Clock Period	Period SSC Off	9.849	10.151	ns		10-5
Positive Duty Cycle (Differential), Negative Duty Cycle (Differential)	Duty Cycle	40	60	%		10-5
	Differential Output Swing	300	—	mV		10-5
Rising Edge Rate	Rising Edge Rate	1.5	4	V/ns		10-5
Falling Edge Rate	Falling Edge Rate	1.5	4	V/ns		10-5
	Jitter	—	150	ps	8,9,10	
SSC	Spread Spectrum	0	0.5	%	11	
CLKOUT_CPUNSSC_P/N						
Rising Edge Rate	Rising Edge Rate	1.3	4	V/ns		10-5
Falling Edge Rate	Falling Edge Rate	1.3	4	V/ns		10-5
	Slew Rate Matching	—	20	%		10-5
Positive Duty Cycle (Differential), Negative Duty Cycle (Differential)	Duty Cycle	45	55	%		10-5
	Jitter	—	6	ps RMS		10-5
CLKOUT_CPUPCIBCLK_P/N, CLKOUT_CPUBCLK_P/N						



Table 10-16. Clock Timings (Sheet 2 of 4)

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
Rising Edge Rate	Rising Edge Rate	1.3	4	V/ns		10-5
Falling Edge Rate	Falling Edge Rate	1.3	4	V/ns		10-5
	Slew Rate Matching	—	20	%		10-5
Positive Duty Cycle (Differential), Negative Duty Cycle (Differential)	Duty Cycle	40	60	%		10-5
	Jitter	—	150	ps		10-5
SMBus/SMLink Clock (SMBCLK, SML[1:0]CLK)						
f _{smb}	Operating Frequency	10	100	kHz		
t ₁₈	High Time	4.0	50	μs	2	10-6
t ₁₉	Low Time	4.7	—	μs		10-6
t ₂₀	Rise Time	—	1000	ns		10-6
t ₂₁	Fall Time	—	300	ns		10-6
SMLink[1,0] (SML[1:0]CLK) (Fast Mode: Refer note 15)						
f _{smb}	Operating Frequency	0	400	kHz		
t _{18_SMLFM}	High Time	0.6	50	μs	2	10-6
t _{19_SMLFM}	Low Time	1.3	—	μs		10-6
t _{20_SMLFM}	Rise Time	—	300	ns		10-6
t _{21_SMLFM}	Fall Time	—	300	ns		10-6
SMLink[1,0] (SML[1,0]CLK) (Fast Mode Plus: Refer note 15)						
f _{smb}	Operating Frequency	0	1000	kHz		
t _{18_SMLFMP}	High Time	0.26	—	μs	2	10-6
t _{19_SMLFMP}	Low Time	0.5	—	μs		10-6
t _{20_SMLFMP}	Rise Time	—	120	ns		10-6
t _{21_SMLFMP}	Fall Time	—	120	ns		10-6
I²C Clock (Standard Mode)						
f _{smb}	Operating Frequency	0	100	kHz		
t _{18_I2CSM}	High Time	4	—	μs	2	10-6
t _{19_I2CSM}	Low Time	4.7	—	μs		10-6
t _{20_I2CSM}	Rise Time	—	1000	ns		10-6
t _{21_I2CSM}	Fall Time	—	300	ns		10-6
I²C Clock (Fast Mode)						
f _{smb}	Operating Frequency	0	400	kHz		
t _{18_I2CFM}	High Time	0.6	—	μs	2	10-6
t _{19_I2CFM}	Low Time	1.3	—	μs		10-6
t _{20_I2CFM}	Rise Time	20	300	ns		10-6
t _{21_I2CFM}	Fall Time	$20 \times \left(\frac{V_{DD}}{V}\right) / 5.5$	300	ns		10-6



Table 10-16. Clock Timings (Sheet 3 of 4)

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
I²C Clock (Fast Mode Plus)						
f _{smb}	Operating Frequency	0	1	MHz		
t18_I2CFMP	High Time	0.26	—	μs	2	10-6
t19_I2CFMP	Low Time	0.5	—	μs		10-6
t20_I2CFMP	Rise Time	—	120	ns		10-6
t21_I2CFMP	Fall Time	$20 \times \frac{V_{DD}}{V}$	120	ns		10-6
I²C Clock (High Speed Mode, Maximum Bus Capacitance (C_B) = 100 pF)						
f _{smb}	Operating Frequency	0	3.4	MHz		
t18_I2CHS1	High Time	60	—	ns	2	10-6
t19_I2CHS1	Low Time	160	—	ns		10-6
t20_I2CHS1	Rise Time	10	40	ns		10-6
t21_I2CHS1	Fall Time	10	40	ns		10-6
I²C Clock (High Speed Mode, Maximum Bus Capacitance (C_B) = 400 pF)						
f _{smb}	Operating Frequency	0	1.7	MHz		
t18_I2CHS2	High Time	120	—	ns	2	10-6
t19_I2CHS2	Low Time	320	—	ns		10-6
t20_I2CHS2	Rise Time	20	80	ns		10-6
t21_I2CHS2	Fall Time	20	80	ns		10-6
HDA_BLK (Intel® High Definition Audio)						
f _{HDA}	Operating Frequency	24.0		MHz		
	Frequency Tolerance	—	100	ppm		
	Input Jitter (refer to Clock Chip Specification)	—	300	ppm		
	High Time (Measured at 0.75 VCC)	18.75	22.91	ns		10-4
	Low Time (Measured at 0.35 VCC)	18.75	22.91	ns		10-4
Suspend Clock (SUSCLK)						
f _{susclk}	Operating Frequency	32		kHz	4	
t39	High Time	9.5	—	μs	4	
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				



Table 10-16. Clock Timings (Sheet 4 of 4)

Symbol	Parameter	Minimum	Maximum	Unit	Notes	Figure
<p>Notes:</p> <ol style="list-style-type: none"> N/A The maximum high time (t18 Max.) provide a simple ensured method for devices to detect bus idle conditions. BCLK Rise and Fall times are measured from 10% VDD and 90% VDD. SUSCLK duty cycle can range from 30% minimum to 70% maximum. Edge rates in a system as measured from 0.8 – 2.0V. 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. Testing condition: 1 kohm Pull-up to VCC, 1 kohm Pull-down and 10 pF Pull-down and 1/2 inch trace. 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. 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. This results in a reflection induced ledge in the middle of the rising edge and will significantly increase measured jitter. Phase jitter requirement: The designated outputs will meet the reference clock jitter requirements from the <i>PCI Express Base Specification</i>. 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 "<i>PCI Express Reference Clock Jitter Measurements</i>". This is not for CLKOUT_ITPXD_P/N. Spread Spectrum (SSC) is referenced to rising edge of the clock. Total of crystal cut accuracy, frequency variations due to temperature, parasitics, load capacitance variations and aging is recommended to be less than 90 ppm. Spread Spectrum (SSC) is referenced to rising edge of the clock. Spread Spectrum (SSC) of 0.25% on CLKOUT_PCIE[15:0] and CLKOUT_PEG_[B:A] is used for WiMAX friendly clocking purposes. 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. Higher fall times are expected at High Speed mode. Validation data shows no functional failures with fall times as low as 9.8 ns and 8.6 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 10-4. Clock Timing

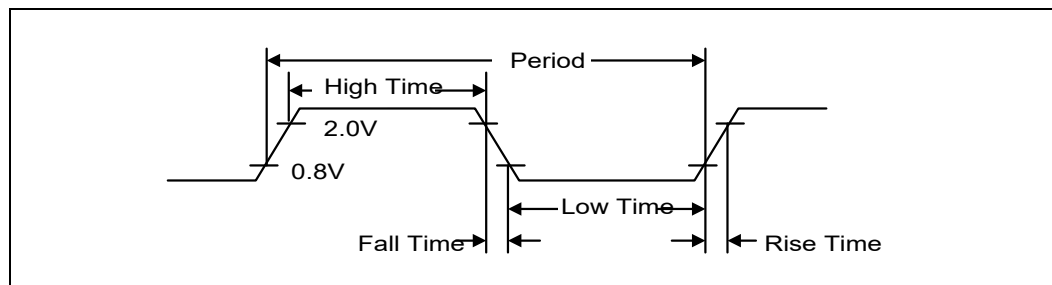


Figure 10-5. Measurement Points for Differential Waveforms

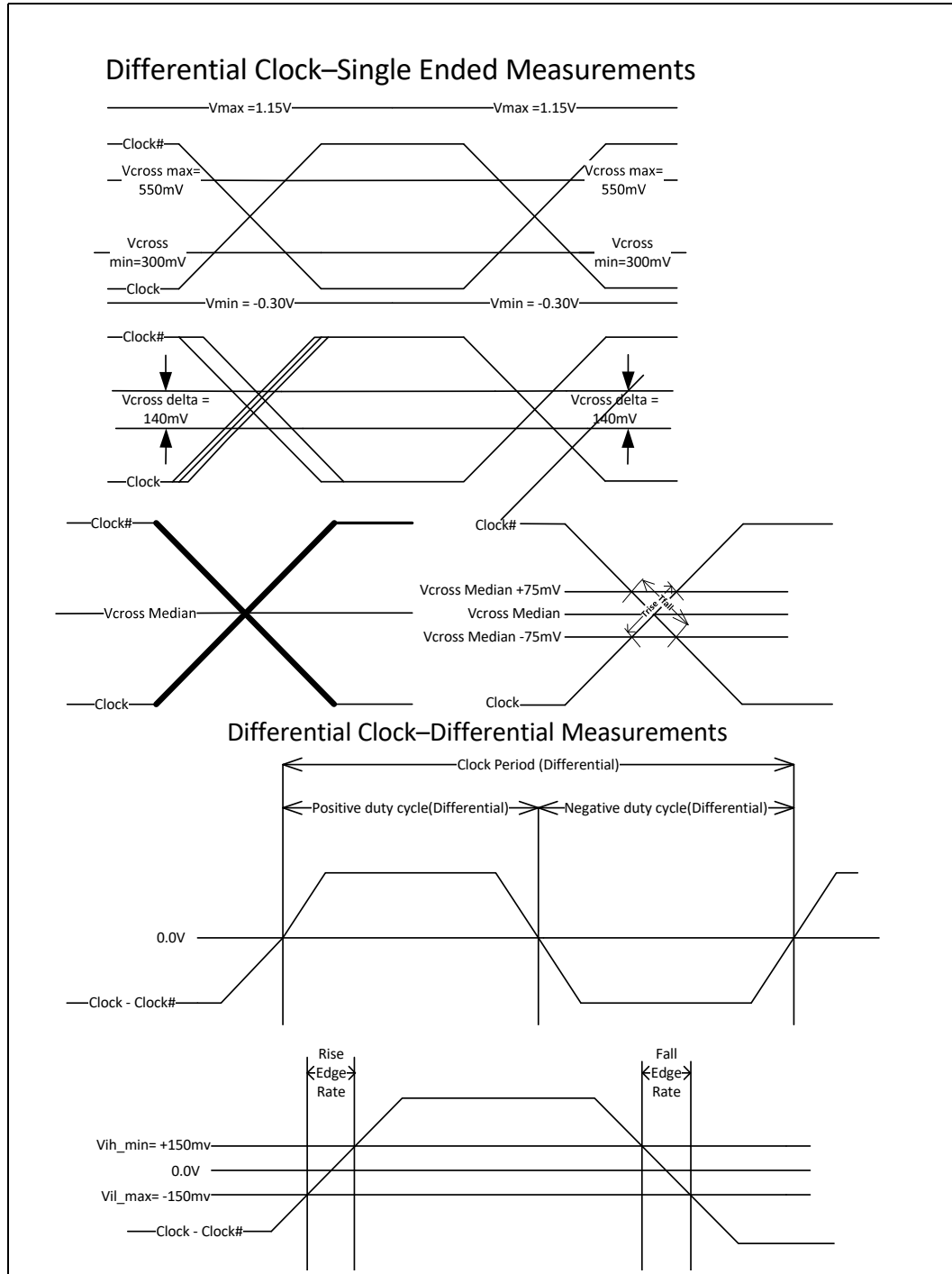




Figure 10-6. I²C, SMBus/SMLink Transaction



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

Table 10-17. USB 2.0 Timing (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
Full-speed Source (Note 7)						
t100	USBPx+, USBPx- Driver Rise Time	4	20	ns	1,6 C _L = 50 pF	10-7
t101	USBPx+, USBPx- Driver Fall Time	4	20	ns	1,6 C _L = 50 pF	10-7
t102	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	-3.5 -4	3.5 4	ns ns	2, 3	10-8
t103	Source SE0 interval of EOP	160	175	ns	4	10-9
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	10-8
t106	EOP Width: Receiver must accept EOP	82	—	ns	4	10-9
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 C _L = 50 pF C _L = 350 pF	10-7
t109	USBPx+, USBPx - Driver Fall Time	75	300	ns	1,6 C _L = 200 pF C _L = 600 pF	10-7
t110	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	-25 -14	25 14	ns ns	2,3	10-8
t111	Source SE0 interval of EOP	1.25	1.50	μs	4	10-9
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	10-8

Table 10-17. USB 2.0 Timing (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t114	EOP Width: Receiver must accept EOP	670	—	ns	4	10-9
t115	Width of SE0 interval during differential transition	—	210	ns		

Notes:

1. Driver output resistance under steady state drive is specified at 28 Ω at minimum and 43 Ω 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 10-18. USB 3.1 Interface Transmit and Receiver Timings

Sym	Parameter	USB 3.1 Gen1(5 GT/s)		USB 3.1 Gen2(10 GT/s)		Units
		Minimum	Maximum	Minimum	Maximum	
UI	Unit Interval	199.94	200.06	99.97	100.03	ps
T _{TX-EYE}	Minimum Transmission Eye Width	0.625	—	0.646	—	UI
P _{U3}	Polling Period U3 State	—	100	—	100	mSs
P _{RX-Detect}	Polling Period Rx Detect	—	100	—	100	mSs

Figure 10-7. USB Rise and Fall Times

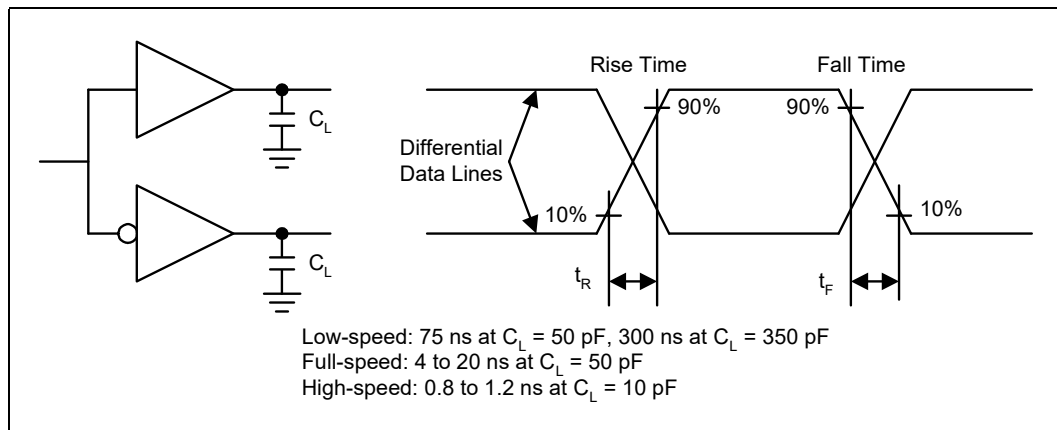




Figure 10-8. USB Jitter

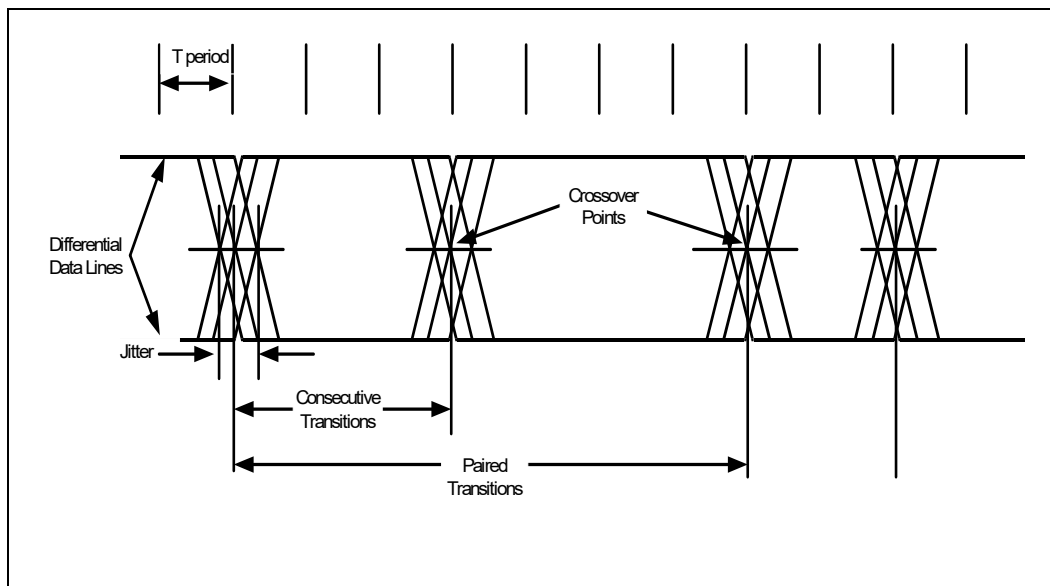


Figure 10-9. USB EOP Width

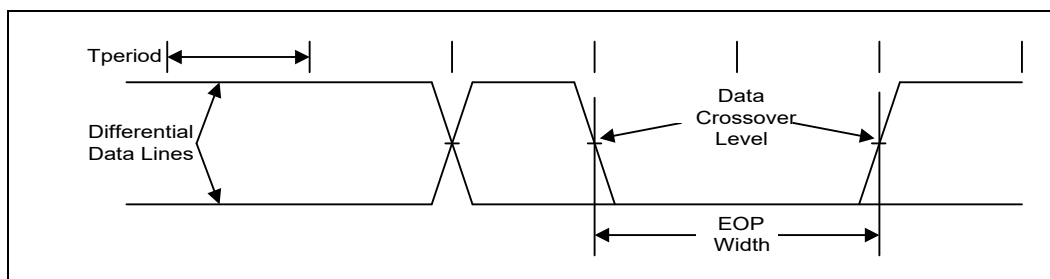


Table 10-19. SATA Interface Timings

Parameter	Minimum	Maximum	Units	Notes
SATA Operating Data Period (6Gb/s)	166.6083	166.6667	ps	
Rise Time	0.2	0.48	UI	1
Fall Time	0.2	0.48	UI	2
TX differential skew	—	20	ps	
COMINIT/COMRESET Transmit Gap Length	310.4	329.6	ns	3
COMWAKE Transmit Gap Length	103.5	109.9	ns	3
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 100mV differential crosspoints of last and first edges of burst				
4. Operating data period during Out-Of-Band burst transmissions				



Table 10-20. SMBus Timings

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t130 _{100 kHz}	Bus Free Time Between Stop and Start Condition	4.7	—	µs		10-6
t131 _{100 kHz}	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4	—	µs		10-6
t132 _{100 kHz}	Repeated Start Condition Setup Time	4.7	—	µs		10-6
t133 _{100 kHz}	Stop Condition Setup Time	4	—	µs		10-6
t134 _{100 kHz}	Data Hold Time	0	—	ns		10-6
t135 _{100 kHz}	Data Setup Time	250	—	ns		10-6
t136	Device Time Out	25	35	ms	1	
t137	Cumulative Clock Low Extend Time (slave device)	—	25	ms	2	10-10
t138	Cumulative Clock Low Extend Time (master device)	—	10	ms	3	10-10
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 10-21. I²C and SMLink Timing (Sheet 1 of 2)

Symbol ²	Parameter	Minimum	Maximum	Units	Notes	Figure
t130 _{SM}	Bus Free Time Between Stop and Start Condition	4.7	—	µs		10-6
t130 _{FM}	Bus Free Time Between Stop and Start Condition	1.3	—	µs		10-6
t130 _{FMP}	Bus Free Time Between Stop and Start Condition	0.5	—	µs		10-6
t131 _{SM}	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4	—	µs		10-6
t131 _{FM}	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.6	—	µs		10-6
t131 _{FMP}	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.26	—	µs		10-6
t131 _{HSM}	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	160	—	ns		10-6
t132 _{SM}	Repeated Start Condition Setup Time	4.7	—	µs		10-6
t132 _{FM}	Repeated Start Condition Setup Time	0.6	—	µs		10-6
t132 _{FMP}	Repeated Start Condition Setup Time	0.26	—	µs		10-6
t132 _{HSM}	Repeated Start Condition Setup Time	160	—	ns		10-6
t133 _{SM}	Stop Condition Setup Time	4	—	µs		10-6
t133 _{FM}	Stop Condition Setup Time	0.6	—	µs		10-6
t133 _{FMP}	Stop Condition Setup Time	0.26	—	µs		10-6



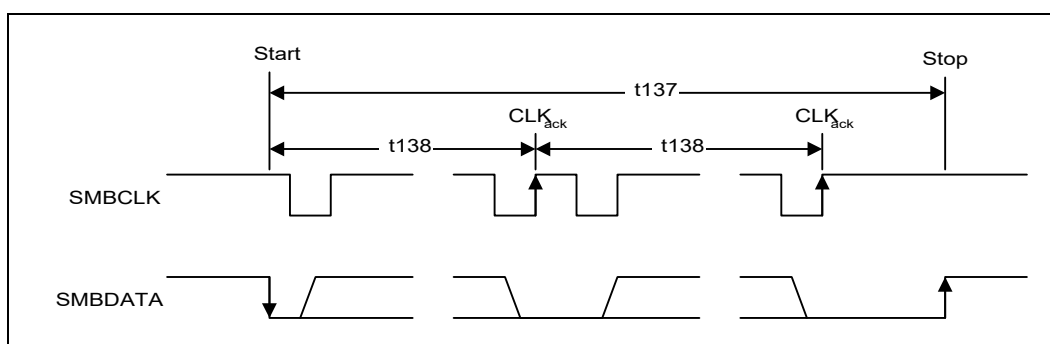
Table 10-21. I²C and SMLink Timing (Sheet 1 of 2)

Symbol ²	Parameter	Minimum	Maximum	Units	Notes	Figure
t133 _{HSM}	Stop Condition Setup Time	160	—	ns		10-6
t134 _{SM}	Data Hold Time	300	—	ns	1	10-6
t134 _{FM}	Data Hold Time	0	—	ns		10-6
t134 _{FMP}	Data Hold Time	0	—	ns		10-6
t135 _{SM}	Data Setup Time	250	—	ns		10-6
t135 _{FM}	Data Setup Time	100	—	ns		10-6
t135 _{FMP}	Data Setup Time	50	—	ns		10-6
t135 _{HSM}	Data Setup Time	10	—	ns		10-6

Notes:

- t134 has a minimum timing for SMLINK is 300 ns.
- 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 10-10. SMBus/SMLink Timeout



Note: SMBCLK also refers to SML[1:0]CLK and SMBDATA also refers to SML[1:0]DATA in Figure 10-6.

Table 10-22. 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.	5.5	13.5	ns		10-11
t144	Time duration for which HDA_SDO is valid after HDA_BCLK edge.	5.5	13.5	ns		10-11
t145	Setup time for HDA_SDI[1:0] at rising edge of HDA_BCLK	20	—	ns		10-11
t146	Hold time for HDA_SDI[1:0] at rising edge of HDA_BCLK	3	—	ns		10-11

Figure 10-11. Intel® High Definition Audio (Intel® HD Audio) Input and Output Timings

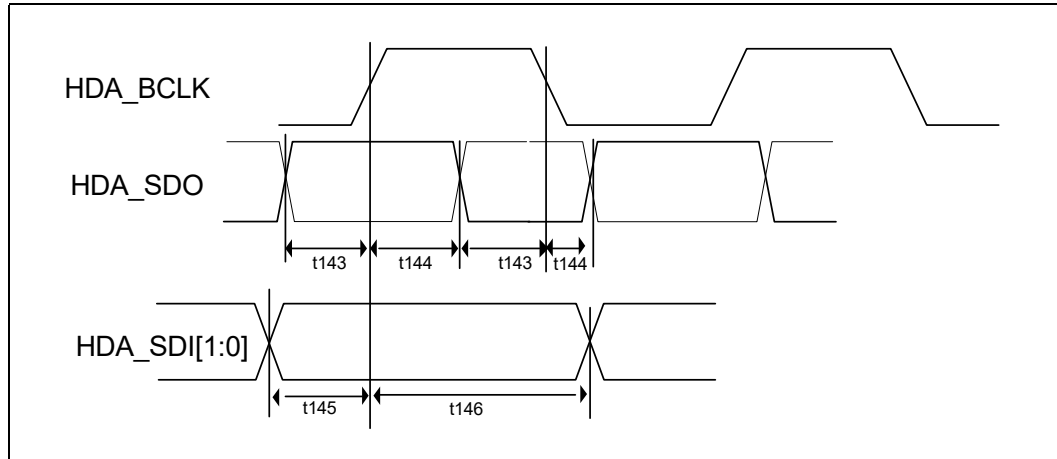


Table 10-23. DMIC Timing

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
	DMIC_DATA[0:1] Setup Time to DMIC_CLK[0:1] Rising	20	—	ns		10-14
	DMIC_DATA[0:1] Hold Time from DMIC_CLK[0:1] Rising	1	—	ns		10-14

Notes: DMIC interface rise and fall times are characterized at the PCH package ball.

Figure 10-12. Valid Delay from Rising Clock Edge

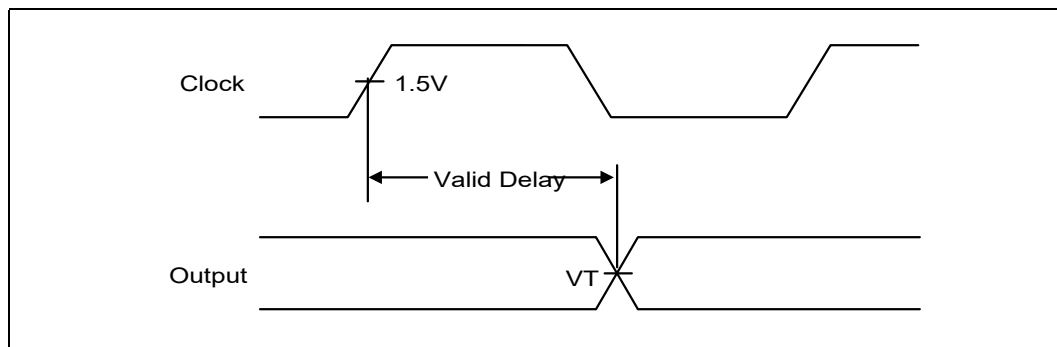


Table 10-24. LPC Timing (24 MHz) (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t150	LAD[3:0] Valid Delay from CLKOUT_LPC[1:0] Rising	1.5	23.5	ns		10-13
t151	LAD[3:0] Output Enable Delay from CLKOUT_LPC[1:0] Rising	2	—	ns		10-16
t152	LAD[3:0] Float Delay from CLKOUT_LPC[1:0] Rising	—	28	ns		10-15
t153	LAD[3:0] Setup Time to CLKOUT_LPC[1:0] Rising	20	—	ns		10-14



Table 10-24. LPC Timing (24 MHz) (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t154	LAD[3:0] Hold Time from CLKOUT_LPC[1:0] Rising	2	—	ns		10-14
t157	LFRAME# Valid Delay from CLKOUT_LPC[1:0] Rising	0	23.5	ns		10-13

Figure 10-13. Valid Delay from Rising Clock Edge

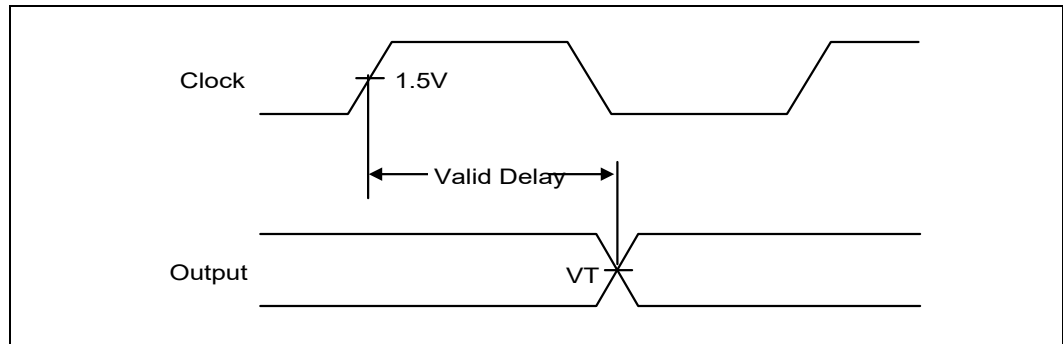


Figure 10-14. Setup and Hold Times

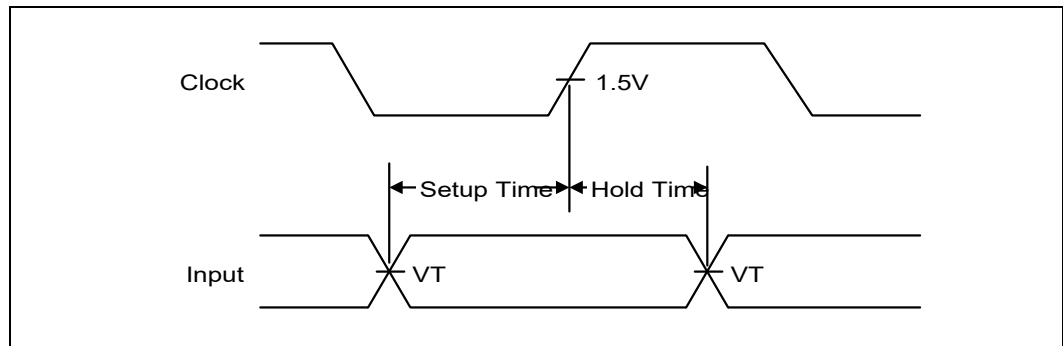


Figure 10-15. Float Delay

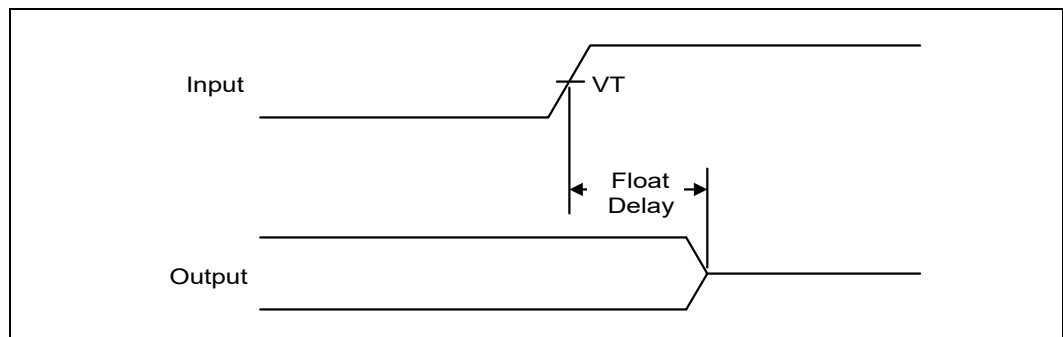


Figure 10-16. Output Enable Delay

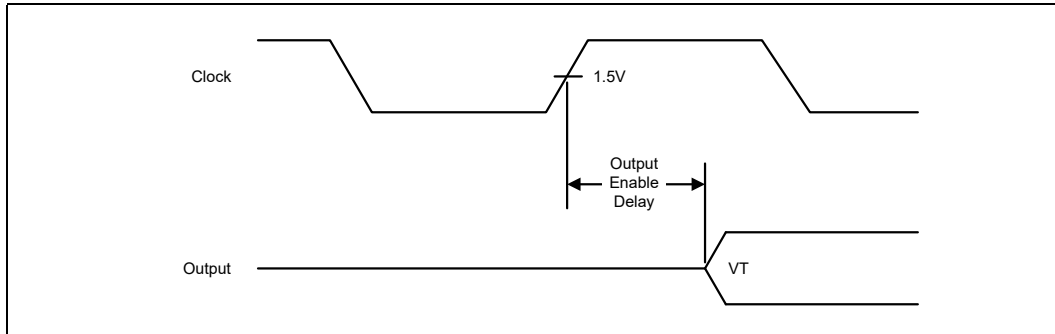


Table 10-25. Miscellaneous Timings

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t160	SERIRQ Setup Time to PCICLK Rising	7	—	ns		10-14
t161	SERIRQ Hold Time from PCICLK Rising	0	—	ns		
t162	GPIO, USB Resume Pulse Width	2	—	RTCCCLK		10-17
t163	SPKR Valid Delay from OSC Rising	—	200	ns		10-13

Figure 10-17. Pulse Width

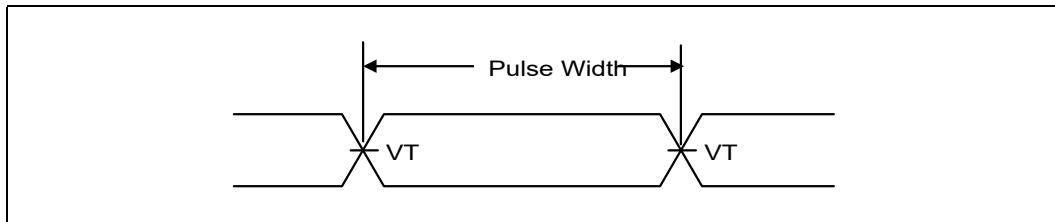


Figure 10-18. PCH Test Load

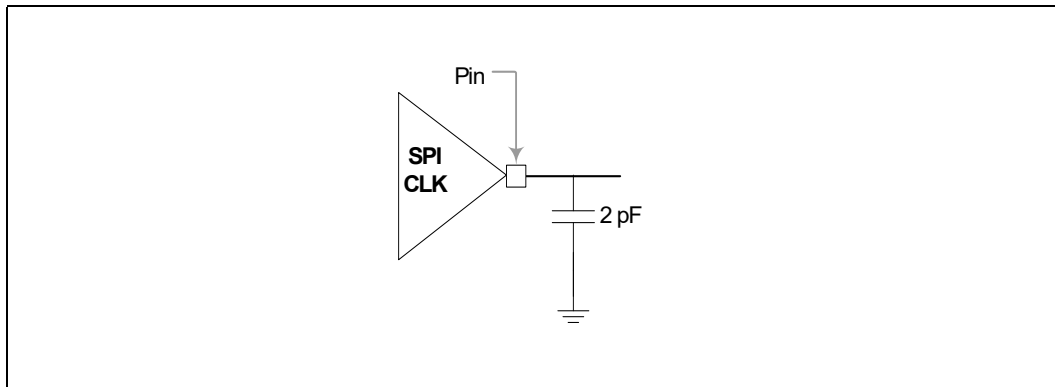


Table 10-26. SPI Timings (20 MHz) (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180a	Serial Clock Frequency	16.8	17.48	MHz	1	



Table 10-26. SPI Timings (20 MHz) (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t183a	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-13	14	ns		10-19
t184a	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	35	—	ns		10-19
t185a	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.5	—	ns		10-19
t186a	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187a	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188a	SPI CLK High time	23.84	—	ns	2	10-19
t189a	SPI CLK Low time	31.84	—	ns	2	10-19

Notes:

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

Table 10-27. SPI Timings (33 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180b	Serial Clock Frequency	29.4	30.6	MHz	1	
t183b	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-8	8	ns		10-19
t184b	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	18	—	ns		10-19
t185b	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.5	—	ns		10-19
t186b	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187b	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188b	SPI CLK High time	16	—	ns	2	10-19
t189b	SPI CLK Low time	16	—	ns	2	10-19

Note:

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

Table 10-28. SPI Timings (50 MHz) (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180c	Serial Clock Frequency	47.04	48.96	MHz	1	
t183c	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-3	6.2	ns		10-19
t184c	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	7	—	ns		10-19
t185c	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1.5	—	ns		10-19
t186c	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19



Table 10-28. SPI Timings (50 MHz) (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t187c	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188c	SPI CLK High time	7.1	—	ns	2, 3	10-19
t189c	SPI CLK Low time	11.17	—	ns	2, 3	10-19
Note: 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 10-29. SPI Touch Timings (20 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180d	Serial Clock Frequency	16.8	17.48	MHz	1	
t183d	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-5	10	ns		10-19
t184d	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	30	—	ns		10-19
t185d	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1	—	ns		10-19
t186d	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187d	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188d	SPI CLK High time	23.84	—	ns	2, 3	10-19
t189d	SPI CLK Low time	31.84	—	ns	2, 3	10-19
Note: 1. Typical clock frequency driven by the PCH is 17.14 MHz. 2. When using 17 MHz mode ensure target flash component can meet t188d and t189d 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 10-30. SPI Touch Timings (33 MHz) 1.8V/3.3V (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180e	Serial Clock Frequency	29.4	30.6	MHz	1	
t183e	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-2.5	7 (3.3 V) 6.5 (1.8 V)	ns		10-19
t184e	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	16.5	—	ns		10-19
t185e	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1	—	ns		10-19
t186e	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187e	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188e	SPI CLK High time	16	—	ns	2, 3	10-19
t189e	SPI CLK Low time	16	—	ns	2, 3	10-19



Table 10-30. SPI Touch Timings (33 MHz) 1.8V/3.3V (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
Note:						
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 10-31. eSPI Timings (33 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180f	Serial Clock Frequency	29.4	30.6	MHz	1	
t183f	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-7.5	7.5	ns		10-19
t184f	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	14	—	ns		10-19
t185f	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1	—	ns		10-19
t186f	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187f	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188f	SPI CLK High time	16	—	ns	2, 3	10-19
t189f	SPI CLK Low time	16	—	ns	2, 3	10-19
Note:						
1. Typical clock frequency driven by the PCH is 30 MHz.						
2. When using 30 MHz mode ensure target flash component can meet t188f and t189f 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 10-32. eSPI Timings (50 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180g	Serial Clock Frequency	47.04	48.96	MHz	1	
t183g	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-6	6	ns		10-19
t184g	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	7.5	—	ns		10-19
t185g	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1	—	ns		10-19
t186g	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187g	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188g	SPI CLK High time	7.84	—	ns	2, 3	10-19
t189g	SPI CLK Low time	11.84	—	ns	2, 3	10-19
Note:						
1. Typical clock frequency driven by the PCH is 48 MHz.						
2. When using 48 MHz mode ensure target flash component can meet t188g and t189g 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 10-33. eSPI Timings (66 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t180h	Serial Clock Frequency	58.8	61.2	MHz	1	
t183h	Tco of SPI MOSI and SPI I/O with respect to serial clock falling edge at the host	-2.5	1	ns		10-19
t184h	Setup of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	5.5	—	ns		10-19
t185h	Hold of SPI MISO and SPI I/O with respect to serial clock falling edge at the host	1	—	ns		10-19
t186h	Setup of SPI CS# assertion with respect to serial clock rising edge at the host	30	—	ns		10-19
t187h	Hold of SPI CS# assertion with respect to serial clock falling edge at the host	30	—	ns		10-19
t188h	SPI CLK High time	8	—	ns	2, 3	10-19
t189h	SPI CLK Low time	8	—	ns	2, 3	10-19

Note:

1. Typical clock frequency driven by the PCH is 60 MHz.
2. When using 60 MHz mode ensure target flash component can meet t188h and t189h 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 10-19. SPI/GSPI/eSPI Timings

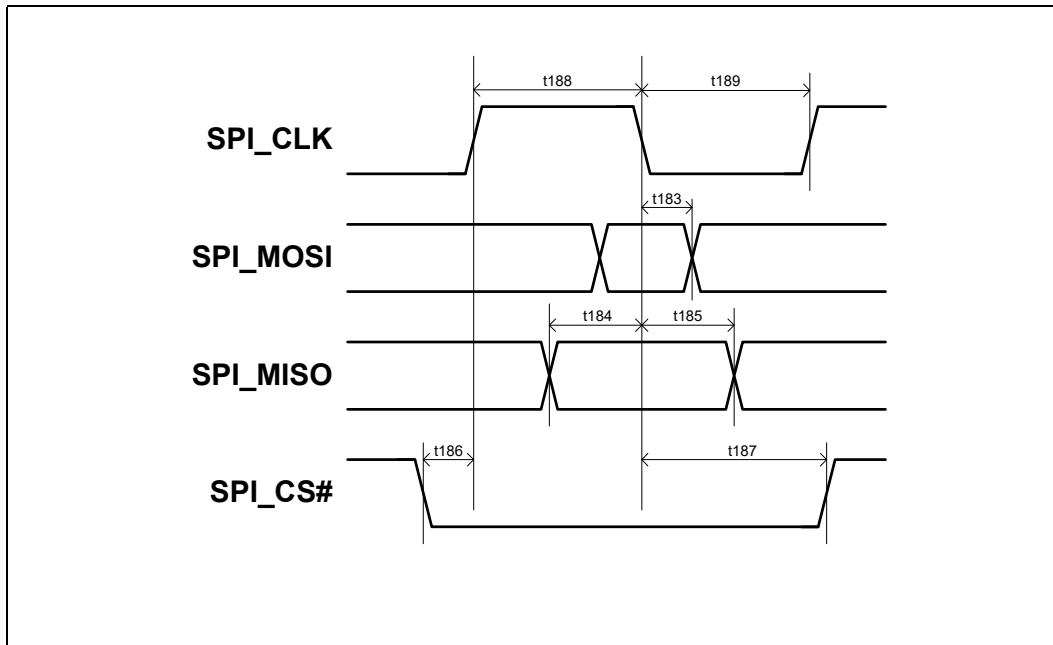


Table 10-34. GSPI Timings (25 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
F	Serial Clock Frequency	—	25	MHz		
t183	Tco of SPI MOSI with respect to serial clock falling edge	-15	7.6	ns		10-19
t184	Setup of SPI MISO and SPI I/O with respect to serial clock rising edge	3.8	—	ns		10-19



Table 10-34. GSPI Timings (25 MHz)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t185	Hold of SPI MISO and SPI I/O with respect to serial clock rising edge	20	—	ns		10-19
t186	Setup of SPI CS# assertion with respect to serial clock rising edge	20	—	ns		10-19
t187	Hold of SPI CS# assertion with respect to serial clock falling edge	20	—	ns		10-19

Table 10-35. Controller Link Receive Timings

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
t190	Single bit time	13	—	ns		10-20
t191	Single clock period	30	—	ns		10-20
t192	Input Signal Slew Rate	0.02857	0.90909	ns	3	10-21
t193	Setup time before CL_CLK	0.9	—	ns		10-20
t194	Hold time after CL_CLK	0.9	—	ns		10-20
V _{IL,AC}	Input low voltage (AC)	—	CL_Vref - 0.08	V	2	
V _{IH,AC}	Input high voltage (AC)	CL_Vref + 0.08	—	V	2	

- Notes:**
1. Measured from (CL_Vref - 50 mV to CL_Vref + 50 mV) at the receiving device side. No test load is required for this measurement as the receiving device fulfills this purpose.
 2. CL_Vref = 0.12*(VCCSus3_3).

Figure 10-20. Controller Link Receive Timings

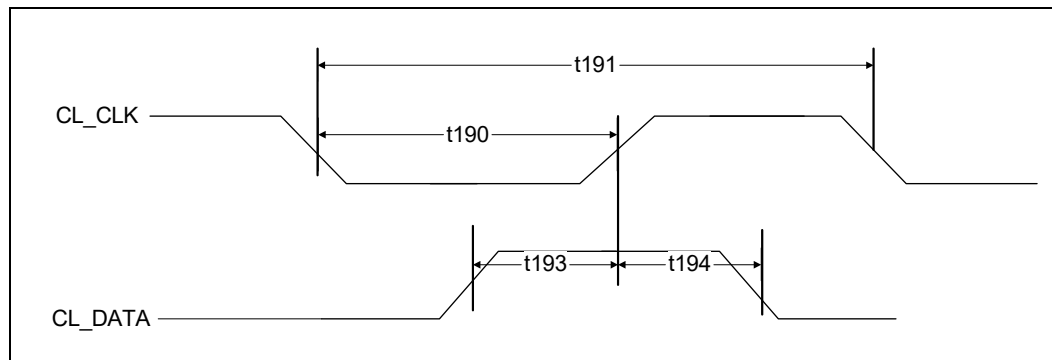


Figure 10-21. Controller Link Receive Slew Rate

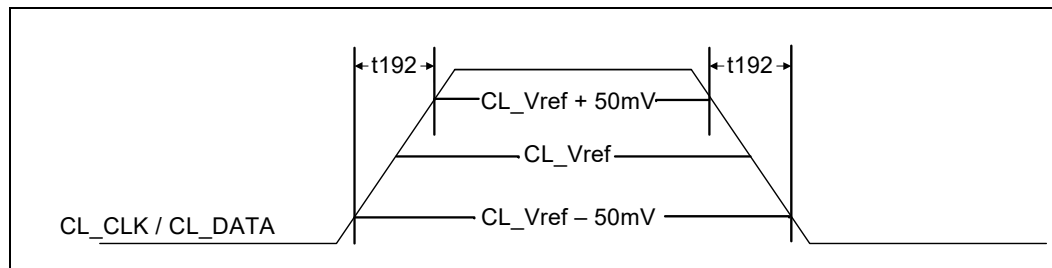




Table 10-36. UART Timings

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

Table 10-37. I²S Timings - Master Mode

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
SCLK						
F _{I2S}	Clock Frequency in (non S0ix)	—	12.288	MHz		
F _{I2S}	Clock Frequency (S0ix)	—	9.6	MHz		
SFRM						
T _{CO}	Clock to Output Delay	—8	15	ns		
RXD						
T _{SU}	Setup Time	40	—	ns		
T _{HD}	Hold Time	1	—	ns		
TXD						
T _{CO}	Clock to Output Delay	—8	15	ns		

Table 10-38. I²S Timing - Slave Mode (non S0ix)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
SCLK						
F _{I2S}	Clock Frequency	—	12.288	MHz		
	Jitter	—	300	ps		
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 10-39. I²S Timing - Slave Mode (S0ix) (Sheet 1 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
SCLK						
F _{I2S}	Clock Frequency	—	9.6	MHz		
	Jitter	—	300	ps		



Table 10-39. I²S Timing - Slave Mode (S0ix) (Continued) (Sheet 2 of 2)

Symbol	Parameter	Minimum	Maximum	Units	Notes	Figure
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		

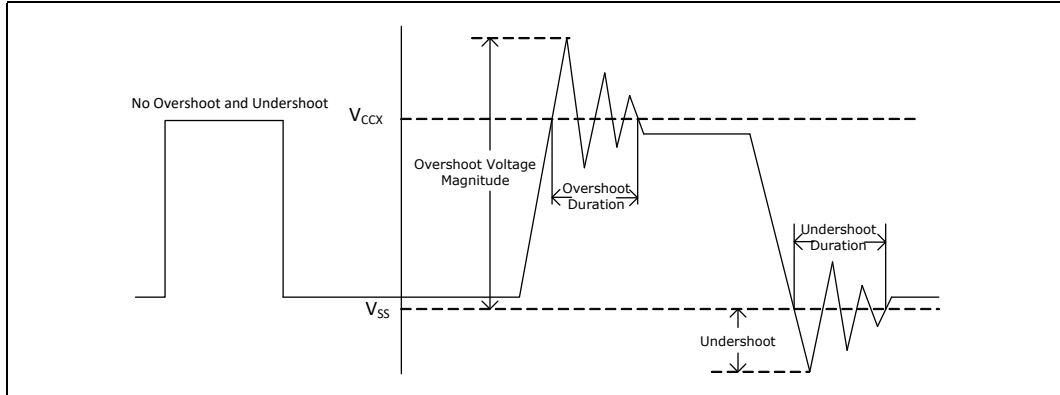
10.5 Overshoot/Undershoot Guidelines

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 in [Table 10-40](#) will ensure reliable I/O performance for the lifetime of the PCH.

Table 10-40. 3.3V Overshoot/Undershoot Specifications

Signal Group	Voltage (VCCx) (V)	Overshoot Voltage Magnitude (V)	Overshoot Duration (nS)	Undershoot Voltage Magnitude (V)	Undershoot Duration (nS)
GPP_A, HDA, GPP_G	1.8	2.01	0.6	-0.15	0.6
		1.97	1.2	-0.12	1.2
	3.3	3.65	2.5	-0.26	2.5
		3.61	5	-0.21	5
GPP_F, GPP_E, SPI	1.8	2.01	0.6	-0.15	0.6
		1.97	1.2	-0.12	1.2
	3.3	3.65	2.5	-0.26	2.5
		3.61	5	-0.21	5
GPP_J	1.8	2.27	0.6	-0.42	0.6
		2.2	1.2	-0.34	1.2
GPP_B, GPP_D, GPP_H, GPP_I, GPP_K, GPP_C, GPD	1.8	1.95	0.6	-0.1	0.6
		1.90	1.2	-0.05	1.2
	3.3	3.51	2.5	-0.11	2.5
		3.45	5	-0.05	5
USB 2.0	3.3	4.356	20.825	-1.056	20.835
Notes:					
1. These specifications are measured at the PCH pin.					
2. Voltage (VCCx) refers to the supply voltage at the pin. Refer to Figure 10-22, "Maximum Acceptable Overshoot/Undershoot Waveform" for pictorial description of allowable overshoot/undershoot magnitude and duration.					

Figure 10-22. Maximum Acceptable Overshoot/Undershoot Waveform



§ §



11 Ballout Definition

This chapter contains the PCH ballout information. [Figure 11-1](#), [Figure 11-2](#), [Figure 11-3](#), and [Figure 11-4](#) show the BGA ballout from a top of package quadrant view. [Table 11-1](#) is a ball list.



Figure 11-1. PCH BGA Ball Map (Top View – Upper Left)

	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25
BG	---	---	DCPRTC	VCCDSW_IP05	VCCDSW_IP05	GPD2_LAN_WA KE#	---	GPD1 / ACPRESEN T	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
BF	---	---	DCPRTC	---	---	GPD0 / BATLOW #	---	GPD4 / SLP_S3#	GPD11 / LANPHY C	SLP_LAN #	---	GPP_A14 / SUS_ST AT# / ESPI_RE SET#	---	GPP_A12 / BM_BUS Y# / ISH_GP6 /SX_EX- IT_HOLD OFF#	GPP_A18 / ISH_GP0	---	GPP_B1 / GSP11_C S1# / TIME_SY NC1	GPP_B3 / CPU_GP 2	GPP_B5 / SRCCLK REQ0#	---	GPP_B16 / GSP10_C LK	---	GPP_C6 / SML1CL K	GPP_C1 / SMBDAT A	GPP_C3 / SMLOCL K
BE	VCCDSW_3P3	VCCDSW_3P3	RTCST #	GPD3 / PWRBTN #	GPD8 / SUSCLK	---	---	GPD5 / SLP_S4#	GPD7	GPD6 / SLP_A#	GPP_A0 / RCIN#	GPP_A5 / LFRAME # / ESPI_CS 0#	---	GPP_A11 / PME# / SD_VDD 2_PWR_ EN#	GPP_A15 / SUSACK #	GPP_A20 / CLKOUT _48	GPP_A16 / VRALERT #	GPP_B6 / SRCCLK REQ1#	GPP_B18 / GSP10_M OST	GPP_B0 / GSP10_C S1#	---	GPP_C7 / SML1DA TA	GPP_C0 / SMBCLK	GPP_C2 / SMBALE RT#	
BD	VCCRTC	---	---	SRTCRTS T#	---	---	---	GPD9 / SLP_WL AN#	---	---	---	GPP_A17 / SD_VDD 1_PWR_ EN# / ISH_GP7	---	---	---	GPP_A19 / ISH_GP1	GPP_B23 / SML1ALE RT# / PCHHOT	---	---	GPP_B21 / GSP11_M ISO	GPP_B17 / GSP10_M ISO	---	---	---	
BC	VCCRTC	---	---	---	---	---	---	GPD10 / SLP_S5#	---	---	---	---	GPP_A13 / SUSWAR N# / SUSPWR DNACK	---	---	---	GPP_B4 / CPU_GP 3	---	---	---	---	GPP_B12 / SLP_S0#	---	---	
BB	---	---	WAKE#	DRAM_R ESET#	---	INTRUDE R#	---	---	---	---	---	GPP_A1 / LAD0 / ESPI_IO 0	---	---	GPP_A9 / CLKOUT LPC0 / ESPI_CL K	GPP_A10 / CLKOUT LPC1	---	---	---	GPP_B8 / SRCCLK REQ3#	---	---	---	GPP_B15 / GSP10_C S0#	---
BA	RTCX1	RTCX2	RSRMRST #	SPIO_IO_3	SPIO_M1 SO	---	---	---	---	---	---	GPP_A4 / LAD3 / ESPI_IO 3	---	GPP_A7 / PIRQA#	---	GPP_A21 / ISH_GP3	---	---	---	GPP_B9 / SRCCLK REQ4#	---	---	---	GPP_B22 / GSP11_M OST	---
AY	---	SPIO_IO_2	SPIO_CS_0#	---	---	---	---	PCH_PW ROK	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
AW	---	SPIO_CS_1#	SPIO_CLK	---	---	---	---	---	DSW_P WROK	---	---	---	GPP_A2 / LAD1 / ESPI_IO 1	---	GPP_A6 / SERIRQ / ESPI_CS 1#	---	---	GPP_A22 / ISH_GP4	---	---	GPP_B14 / SPKR	---	---	GPP_B19 / GSP11_C S0#	---
AV	---	---	GPP_F15 / USB2_O C4#	GPP_F20 / eDP_BKL TEN	---	GPP_F19 / eDP_VD DEN	GPP_F18 / USB2_O C7#	---	---	---	---	---	GPP_A3 / LAD2 / ESPI_IO 2	---	---	GPP_A23 / ISH_GP5	---	GPP_A8 / CLKRUN #	---	---	GPP_B13 / PLTRST#	---	---	---	---
AU	---	GPP_F21 / eDP_BKL TCTL	GPP_F13 / SATA_S- DATAOU T0	GPP_F12 / SATA_S- DATAOU T1	---	---	---	---	SPIO_M0 SI	---	---	---	---	---	---	---	---	---	---	---	---	---	---	GPP_B20 / GSP11_C LK	---
AT	GPP_F22 / DDPF_C- TRCLK	---	---	---	---	VCCPR1 M_3P3	---	---	---	SPIO_CS_2#	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
AR	---	GPP_F11 / SATA_SL OAD	GPP_F6 / SATA_DE VSLF4	---	---	---	---	GPP_F10 / SATA_SC LOCK	---	---	---	---	GPP_F17 / USB2_O C6#	---	GPP_F16 / USB2_O C5#	---	---	GPP_B7 / SRCCLK REQ2#	---	---	---	---	---	---	---
AP	---	GPP_F5 / SATA_DE VSLF3	GPP_F9 / SATA_DE VSLF7	---	---	---	---	---	GPP_F14 / PS_ON#	---	---	---	---	---	---	---	---	---	---	---	GPP_B11 / I2S_MCL K	---	---	VCCPGP PBC	---
AN	---	---	GPP_F0 / SATA_XPC IE3 / SATAGP3	GPP_F7 / SATA_DE VSLP5	---	VCCSPI	---	---	---	GPP_F23 / DDPF_C- TRLDATA	---	---	GPP_F8 / SATA_DE VSLP6	---	TP	---	VCCPGP PA	---	---	GPP_B10 / SRCCLK REQ5#	---	---	---	VCCPGP PBC	---
AM	---	GPP_F4 / SATA_XPC IE7 / SATAGP7	GPP_F3 / SATA_XPC IE6 / SATAGP6	GPP_F1 / SATA_XPC IE4 / SATAGP4	GPP_E7 / CPU_GP 1	---	GPP_F2 / SATA_XPC IE5 / SATAGP5	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
AL	---	GPP_E4 / SATA_DE VSLP0	GPP_E3 / CPU_GP 0	---	---	---	---	---	GPP_E12 / USB2_O C3#	GPP_E10 / USB2_O C1#	---	---	---	---	TP	---	---	---	---	---	---	---	---	---	---
AK	---	GPP_E8 / SATALED #	GPP_E2 / SATA_XPC IE2 / SATAGP2	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
AJ	---	---	GPP_H23 / TIME_SY NC0	GPP_H1 / SML4DA TA	---	GPP_E11 / USB2_O C2#	GPP_E1 / SATA_XPC IE1 / SATAGP1	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
AH	---	GPP_H2 17 / ISH_I2C 1_SDA	GPP_H22 / ISH_I2C 1_SCL	GPP_H1 97 / ISH_I2C 0_SDA	---	---	---	---	GPP_E0 / SATA_XPC IE0 / SATAGP0	GPP_E6 / SATA_DE VSLF2	---	---	---	GPP_E9 / USB2_O C0#	GPP_E5 / SATA_DE VSLF1	---	---	---	---	---	---	---	---	---	---
AG	---	---	---	GPP_H2 07 / ISH_I2C 0_SCL	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPR1 M_IP05	---	---	---	---	---
AF	---	GPP_H3 / SRCCLK REQ9#	GPP_H13 / SML3CL K	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPR1 M_IP05	VCCPR1 M_IP05	---	---	VCCPR1 M_IP05	---
AE	---	GPP_H1 0 / SML2CL K	GPP_H0 / SRCCLK REQ8#	---	---	GPP_H18 / SML4ALE RT#	GPP_H1 6 / SML4CL K	---	GPP_H2 / SRCCLK REQ8#	---	GPP_H6 / SRCCLK REQ12#	---	---	VCCPGP PEF	VCCPGP PEF	---	---	---	---	---	---	---	---	---	---
AD	---	GPP_H1 4 / SML3DA TA	GPP_H11 / SML2DA TA	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPR1 M_IP05	VCCPR1 M_IP05	---	VCCPR1 M_IP05	VCCPR1 M_IP05	---



Figure 11-2. PCH BGA Ball Map (Top View – Upper Right)

	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
BG	---	---	---	---	---	---	---	---	---	---	---	HDA_SY NC / I2S0_SF_RX	---	---	---	---	GPP_G4 / SD_DAT_A3	---	CNV_WT_DIN	VCCDPH_Y_IP24	---	---	---	---	---	
BF	GPP_C5 / SMLDALE_RT#	GPP_C16 / I2C0_SD_A	---	GPP_C18 / I2C1_SD_A	---	GPP_D0 / SPI1_CS# / SBK0	GPP_D3 / SPI1_MO_S1 / SBK3 / BK3	GPP_D1 / ISH_UAR_TO_TXD	---	GPP_D2 / DMIC_D_ATAD / SNDW4_DATA	GPP_D1 / ISH_UAR_TO_CTS# / CNV_WC_EN	---	HDA_SD_O7 / I2S0_TX_D	---	HDA_SD_I17 / I2S1_RX_D	GPP_G3 / SD_DAT_A2	GPP_G2 / SD_DAT_A1	---	CNV_WT_DP	---	---	---	---	---	---	---
BE	GPP_C4 / SMLDATA	GPP_C8 / UART0A_RXD	---	GPP_C19 / I2C1_SC_L	GPP_C21 / UART2_T_XD	GPP_D1 / SPI1_CL_K / SBK1 / BK1	GPP_D2 / SPI1_MO_S0 / SBK2 / BK2	GPP_D1 / ISH_UAR_TO_RXD	GPP_D5 / I2S2_SF_RXV / CNV_RF_RESET#	GPP_D4 / ISH_I2C_2_SDA / I2C3_SD_A / SBK4 / BK4	GPP_D2 / ISH_I2C_2_SCL / I2C1_SC_L	---	I2S1_TX_D7 / SNDW2_DATA	HDA_SD_I07 / I2S0_RX_D	HDA_RS_T# / I2S1_SC_LK	GPP_G1 / SD_DAT_A0	GPP_G5 / SD_CD#	---	CNV_WT_D0N	SD_IP8_RCOMP	SD_3P3_RCOMP	CNV_WR_CLKP	GPP1_R_COMP_1_P8	GPP1_R_COMP_1_P8	---	
BD	---	---	---	GPP_C15 / UART1_CTS# / ISH_UAR_T1_CTS#	GPP_C20 / UART2_RXD	---	---	GPP_D2 / SPI1_IO_2	GPP_D1 / DMIC_CL_K0 / SNDW4_CLK	---	---	---	I2S1_SF_RM7 / SNDW2_CLK	HDA_BC_LK7 / I2S0_SC_LK	---	---	GPP_G6 / SD_CLK	CNV_WT_D0P	---	---	CNV_WR_CLKN	---	---	GPP1_R_COMP_1_P8	---	
BC	---	---	GPP_C17 / I2C0_SC_L	---	---	---	---	GPP_D2 / SPI1_IO_3	---	---	---	---	---	---	---	---	---	---	---	CNV_WT_CLKN	---	---	---	---	RSVD	
BB	GPP_C9 / UART0A_TXD	---	---	---	GPP_D1 / ISH_SPI_CLK / GSP12_C_LK	---	---	---	GPP_D1 / ISH_SPI_MISO / GP_BSS_B_CLK / GSP12_M_ISO	---	VCCPDA	---	---	VCCPRI_M_IP6	---	---	---	VCCPRI_M_3P3	CNV_WT_CLKP	---	CNV_WR_D0P	CNV_WR_D0N	---	---	---	
BA	GPP_C10 / UART0A_RTS#	---	---	---	GPP_D9 / ISH_SPI_CS# / GSP12_C_SD#	---	---	GPP_D8 / I2S2_TX_D7 / MODEM_CLKREQ	---	---	---	---	---	---	---	---	---	---	---	---	---	GPP_J6 / CNV_RG_I_DT / UART0B_TXD	CNV_WR_DIN	CNV_WR_DIP	CNV_WT_RCOMP	
AY	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPRI_M_3P3	---	---	---	---	---	GPP_J1 / CPU_C10_GATE#	GPP_J5 / CNV_BR1_RSP / UART0B_RXD	---	---	
AW	GPP_C14 / UART1_RTS# / ISH_UAR_T1_RTS#	---	---	GPP_C22 / UART2_RTS#	---	---	GPP_D7 / I2S2_RX_D	---	---	GPP_D1 / DMIC_CL_K1 / SNDW3_CLK	---	---	GPP_G0 / SD_CMD	---	---	VCCPRI_M_3P3	---	---	---	---	---	---	GPP_J2	GPP_J8 / CNV_M_FUART2_RXD	---	
AV	---	---	---	GPP_C23 / UART2_CTS#	---	---	GPP_D8 / I2S2_SC_LK	---	---	GPP_D1 / DMIC_D_ATAI / SNDW3_DATA	---	---	GPP_G7 / SD_WP	---	---	---	---	GPP_J10	GPP_J0 / CNV_PA_BLANKING	---	GPP_J4 / CNV_BR1_DT / UART0B_RTS#	GPP_J7 / CNV_RG_I_RSP / UART0B_CTS#	---	---	---	
AU	GPP_C12 / UART1_RXD / ISH_UAR_T1_RXD	---	---	---	---	---	---	---	---	---	---	---	---	---	---	GPP_J9 / CNV_M_FUART2_TXD	---	---	---	---	---	CL_RST#	SYS_PW_R0K	SYS_RE_SET#	---	
AT	---	---	---	---	---	---	---	---	---	---	---	---	---	---	GPP_J3	---	---	---	GPP_J0 / DDPB_H_PD07 / DISP_MI_SCD	CL_DATA	---	---	---	---	---	
AR	---	---	---	---	---	---	GPP_D1 / ISH_UAR_TO_RTS# / GSP12_C_S1# / CNV_WF_EN	---	---	VCCPRI_M_IP8	---	GPP_J11 / A4WP_P_RESENT	---	---	---	---	---	GPP_J6 / DDPB_C_TRLDATA	---	---	---	---	GPP_J10 / DDPB_C_TRLDATA	CL_CLK	---	
AP	GPP_C11 / UART0A_CTS#	---	---	GPP_C13 / UART1_TXD / ISH_UAR_T1_TXD	---	---	---	---	---	---	---	---	---	---	---	GPP_J2 / DDPD_H_PD2 / DISP_MI_SC2	---	---	---	---	---	---	GPP_J11 / M2_SKT_2_CFG0	GPP_J12 / M2_SKT_2_CFG1	---	
AN	VCCPDP	---	---	VCCPDP_PG_3P3	---	---	GPP_D1 / ISH_SPI_MOSI / GP_BSS_B_D1 / GSP12_M_OSI	---	---	VCCPRI_M_IP8	---	GPP_J17 / DDPB_C_TRLCLK	---	---	GPP_J11 / DDPC_H_PD1 / DISP_MI_SC1	---	---	---	GPP_J4 / EDP_HP_D7 / DISP_MI_SC4	---	GPP_J13 / M2_SKT_2_CFG2	HDACPU_SDI	---	---	---	
AM	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	GPP_J14 / M2_SKT_2_CFG3	---	PRDY#	CPU_TR_ST#	HDACPU_SCLK	HDACPU_SDO	---	---	---	
AL	---	---	---	---	---	---	---	---	GPP_J3 / DDPB_H_PD3 / DISP_MI_SC3	---	GPP_J5 / DDPB_C_TRLCLK	---	---	---	GPP_J8 / DDPC_C_TRLDATA	GPP_J9 / DDPD_C_TRLCLK	---	---	---	---	---	---	ITP_PMO_DE	PREQ#	---	
AK	---	VCCDPH_Y_IP24	VCCDPH_Y_IP24	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	TRIGGER_OUT	TRIGGER_IN	---	
AJ	---	VCCDPH_Y_IP24	VCCDPH_Y_IP24	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
AH	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
AG	---	---	---	---	VCCPRI_M_IP9	VCCPRI_M_IP8	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
AF	---	VCCPRI_M_IP05	---	---	VCCPH_VLDO_1_P8	VCCPH_VLDO_1_P8	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
AE	---	---	---	---	---	---	---	VCCPRI_M_IP05	---	CLKOUT_PCIE_P_2	CLKOUT_PCIE_N_2	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
AD	---	VCCPRI_M_IP05	---	VCCPRI_M_IP05	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	



Figure 11-3. PCH BGA Ball Map (Top View – Lower Left)

	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	
AC	---	GPP_H1 SRCCLK REQ7#	GPP_H15 SML3ALE RT#	---	---	GPP_H8 SRCCLK REQ14#	GPP_H9 SRCCLK REQ15#	---	GPP_H4 SRCCLK REQ10#	---	GPP_H5 SRCCLK REQ11#	---	---	VCCPGP PHK	VCCPGP PHK	---	---	---	---	---	---	---	---	---	---	---
AB	---	GPP_H7 SRCCLK REQ13#	GPP_H12 SML2ALE RT#	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPRI M_IP05	---	VCCPRI M_IP05	VCCPRI M_IP05	---	---	
AA	---	---	---	GPP_K15 GSXSRE SET#	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
Y	---	GPP_K13 GSXSLO AD	GPP_K16 GSXCLK	GPP_K12 GSXDOU T	---	---	---	---	PCIE24 RXP	PCIE24 RXN	---	---	---	RSVD	RSVD	---	---	---	---	---	---	---	---	---	---	
W	---	---	GPP_K11	GPP_K14 GSXDIN	---	PCIE23 RXP	PCIE23 RXN	---	---	---	---	---	---	---	---	---	---	---	---	VCCPRI M_MPHY IP05	---	---	---	---	---	
V	---	GPP_K10	GPP_K9	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	VCCPRI M_IP05	VCCPRI M_IP05	---	VCCPRI M_IP05	VCCPRI M_IP05	VCCPRI M_IP05	
U	---	GPP_K2	GPP_K3	---	---	---	---	---	PCIE22 RXP	PCIE22 RXN	---	---	---	RSVD	---	RSVD	---	---	---	---	VCCPRI M_IP05	---	---	---	VCCPRI M_IP05	
T	---	GPP_K19 / SMI#	GPP_K18 / NMI#	GPP_K20	GPP_K21	---	PCIE21 RXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
R	---	---	GPP_K17 / ADR COMPLETE	GPP_K7	---	PCIE21 RXP	---	---	---	PCIE18 RXP / SATA5_R XP	---	---	---	PCIE20 RXP / SATA7_R XP	---	PCIE20 RXN / SATA7_R XN	---	RSVD	---	---	---	---	---	---	---	
P	---	GPP_K8	GPP_K6	---	---	---	---	---	PCIE18 RXN / SATA5_R XN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
N	---	GPP_K4	GPP_K5	---	---	---	---	PCIE19 RXP / SATA6_R XP	---	---	---	---	---	---	---	---	---	RSVD	---	---	---	---	---	---	---	
M	---	---	---	GPP_K23 / IMGCLK OUT1	---	PCIE19 RXN / SATA6_R XN	---	---	---	PCIE16 RXP / SATA3_R XP	---	---	---	---	---	---	---	---	---	---	---	RSVD	---	---	RSVD	
L	---	GPP_K22 / IMGCLK OUT0	GPP_K0	GPP_K1	---	---	---	---	PCIE16 RXN / SATA3_R XN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	RSVD	
K	---	---	VCCMPH Y_SENSE	VSSMPH Y_SENSE	---	PCIE17 RXP / SATA4_R XP	PCIE17 RXN / SATA4_R XN	---	---	---	---	---	---	PCIE10 RXN	---	---	DMI0_R XN	---	DMI2_R XN	---	---	RSVD	---	---	---	
J	---	---	---	---	---	---	---	---	PCIE12 RXP / SATA1A RXP	---	---	---	---	---	---	---	PCIE10 RXP	---	DMI0_R XP	---	---	DMI2_R XP	---	---	---	
H	---	PCIE22 TXP	PCIE22 TXN	---	---	---	---	---	PCIE12 RXN / SATA1A RXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	
G	PCIE23 TXN	PCIE23 TXP	PCIE21 TXN	PCIE24 TXN	PCIE24 TXP	---	---	---	---	---	---	---	---	PCIE11 RXN / SATA0A RXN	---	PCIE9_R XN	---	---	DMI1_R XN	---	---	DMI3_R XN	---	---	RSVD	
F	---	---	---	PCIE21 TXP	---	PCIE15 RXN / SATA2_R XN	---	---	---	---	PCIE11 RXP / SATA0A RXP	---	---	---	PCIE9_R XP	---	DMI1_R XP	---	---	---	---	DMI3_R XP	---	---	RSVD	
E	VCCAM- PHYLL IP05	---	---	---	PCIE15 RXP / SATA2_R XP	---	---	---	---	---	---	---	---	---	---	PCIE12 TXP / SATA1A TXP	---	---	---	---	---	RSVD	---	---	---	
D	VCCAM- PHYLL IP05	---	---	PCIE14 RXN / SATA1B RXN	---	---	PCIE19 TXP / SATA6_T XP	PCIE18 TXP / SATA5_T XP	---	---	PCIE14 TXP / SATA1B TXP	PCIE12 TXN / SATA1A TXN	---	---	---	---	PCIE9_T XP	---	---	---	---	---	RSVD	---	---	
C	VCCAM- PHYLL IP05	---	PCIE14 RXP / SATA1B RXP	PCIE13 RXP / SATA0B RXP	PCIE13 RXN / SATA0B RXN	PCIE19 TXN / SATA6_T XN	---	PCIE18 TXN / SATA5_T XN	PCIE16 TXP / SATA3_T XP	PCIE15 TXP / SATA2_T XP	PCIE14 TXN / SATA1B TXN	PCIE13 TXP / SATA0B TXP	---	PCIE11 TXP / SATA0A TXP	PCIE10 TXN	PCIE9_T XN	DMI0_TX XN	DMI1_TX XN	DMI2_TX XN	---	DMI3_TX XN	---	---	RSVD	RSVD	
B	---	---	---	---	---	PCIE20 TXP / SATA7_T XP	---	PCIE17 TXP / SATA7_T XP	PCIE16 TXN / SATA3_T XN	PCIE15 TXN / SATA2_T XN	---	PCIE13 TXN / SATA0B TXN	---	---	PCIE11 TXN / SATA0A TXN	PCIE10 TXP	---	DMI0_TX XP	DMI1_TX P	DMI2_TX P	---	DMI3_TX P	---	RSVD	RSVD	
A	---	---	---	---	---	PCIE20 TXN / SATA7_T XN	---	PCIE17 TXN / SATA4_T XN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	RSVD	

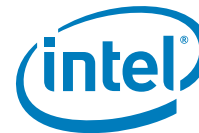


Figure 11-4. PCH BGA Ball Map (Top View – Lower Right)

	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
AC	---	---	---	---	---	---	---	---	---	CLKOUT_PCIE_P_8	CLKOUT_PCIE_N_8	---	---	CLKOUT_PCIE_P_10	---	CLKOUT_PCIE_N_10	---	CLKOUT_PCIE_N_12	CLKOUT_PCIE_P_12	---	---	CLKOUT_PCIE_P_4	CLKOUT_PCIE_N_4	---	
AB	---	VCCPRI_M_1P05	VCCPRI_M_1P05	---	VCCPRI_M_1P05	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_PCIE_P_5	CLKOUT_PCIE_N_5	---	
AA	---	VCCPRI_M_1P05	VCCPRI_M_1P05	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_PCIE_N_13	
Y	---	---	---	---	---	---	---	---	---	---	RSVD	RSVD	---	---	---	---	---	---	---	---	---	CLKOUT_ITPXD_P	CLKOUT_ITPXD_N	CLKOUT_PCIE_P_13	
W	---	VCCDUS_8_1P05	VCCDUS_8_1P05	---	VCCA_S_RC_1P05	VCCA_S_RC_1P05	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_PCIE_N_7	CLKOUT_PCIE_P_7	---	---	CLKOUT_PCIE_N_6	CLKOUT_PCIE_P_6	---	
V	---	VCCPRI_M_3P3	---	---	---	VCCA_B_CLK_1P05	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_PCIE_P_15	CLKOUT_PCIE_N_15	---	
U	---	---	---	---	---	---	---	---	---	---	---	RSVD	---	XTAL_IN	XTAL_OUT	---	---	---	---	---	---	---	CLKOUT_PCIE_P_9	CLKOUT_PCIE_N_9	
T	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	XCLK_BI_ASREF	CLKOUT_PCIE_P_14	CLKOUT_PCIE_N_14	
R	RSVD	---	---	PCIE2_RXN / USB31_8_RXN	---	---	PCIE4_RXP / USB31_10_RXP	---	---	RSVD	---	RSVD	---	---	USB2N_11	---	---	---	CLKIN_XTAL	---	---	---	---	---	
P	RSVD	---	---	PCIE2_RXP / USB31_8_RXP	---	---	---	---	---	---	---	---	---	---	---	USB2P_11	---	---	---	---	---	VCCA_XTAL_1P05	VCCA_XTAL_1P05	---	
N	---	---	---	---	---	---	PCIE4_RXN / USB31_10_RXN	---	---	---	USB2P_2	---	USB2N_2	---	---	---	---	---	---	---	---	---	USB2N_13	USB2P_13	
M	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_4	---	---	---	---	---	---	---	USB2N_5	
L	PCIE7_RXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2P_4	---	---	---	---	USB2N_7	USB2P_7	USB2P_5	
K	---	---	---	PCIE6_RXN	---	---	PCIE3_RXN / USB31_9_RXN	---	---	---	USB31_4_RXN	---	---	USB31_5_RXP	---	---	---	---	---	---	---	---	USB2N_3	USB2P_3	---
J	PCIE7_RXP	---	---	PCIE6_RXP	---	---	PCIE3_RXP / USB31_9_RXP	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_1	USB2P_1
H	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_10	USB2P_10
G	PCIE8_RXP	---	---	---	PCIE5_RXP	---	---	PCIE1_RXN / USB31_7_RXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_12
F	PCIE8_RXN	---	---	---	PCIE5_RXN	---	---	PCIE1_RXP / USB31_7_RXP	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_8
E	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	USB2N_14
D	---	---	---	PCIE6_TXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_CPUN_SC_P
C	PCIE8_TXP	PCIE7_TXN	---	PCIE6_TXP	PCIE4_TXP / USB31_10_TXP	PCIE3_TXP / USB31_9_TXP	PCIE2_TXP / USB31_8_TXP	USB31_6_TXN	USB31_6_TXP	USB31_5_TXN	USB31_4_TXP	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_CPUN_SC_N
B	PCIE8_TXN	PCIE7_TXP	---	PCIE5_TXN	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_CPUN_SC_P
A	---	---	PCIE5_TXP	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	CLKOUT_CPUN_SC_N



Table 11-1. PCH BGA Ballout (Sheet 1 of 24)

Ball Name	Ball #
CL_CLK	AR2
CL_DATA	AT5
CL_RST#	AU4
CLKIN_XTAL	R6
CLKOUT_CPUBCLK_N	C8
CLKOUT_CPUBCLK_P	B8
CLKOUT_CPUNSSC_N	C6
CLKOUT_CPUNSSC_P	D7
CLKOUT_CPUPCIBCLK_N	B6
CLKOUT_CPUPCIBCLK_P	A6
CLKOUT_ITPXD_P	Y3
CLKOUT_ITPXD_P	Y4
CLKOUT_PCIE_N0	AJ6
CLKOUT_PCIE_N1	AH9
CLKOUT_PCIE_N10	AC9
CLKOUT_PCIE_N11	AE9
CLKOUT_PCIE_N12	AC7
CLKOUT_PCIE_N13	AA1
CLKOUT_PCIE_N14	T2
CLKOUT_PCIE_N15	V2
CLKOUT_PCIE_N2	AE14
CLKOUT_PCIE_N3	AE6
CLKOUT_PCIE_N4	AC2
CLKOUT_PCIE_N5	AB2
CLKOUT_PCIE_N6	W4
CLKOUT_PCIE_N7	W7
CLKOUT_PCIE_N8	AC14
CLKOUT_PCIE_N9	U2
CLKOUT_PCIE_P0	AJ7
CLKOUT_PCIE_P1	AH10
CLKOUT_PCIE_P10	AC11
CLKOUT_PCIE_P11	AE11
CLKOUT_PCIE_P12	AC6
CLKOUT_PCIE_P13	Y2
CLKOUT_PCIE_P14	T1
CLKOUT_PCIE_P15	V3
CLKOUT_PCIE_P2	AE15
CLKOUT_PCIE_P3	AE7
CLKOUT_PCIE_P4	AC3

Table 11-1. PCH BGA Ballout (Sheet 2 of 24)

Ball Name	Ball #
CLKOUT_PCIE_P5	AB3
CLKOUT_PCIE_P6	W3
CLKOUT_PCIE_P7	W6
CLKOUT_PCIE_P8	AC15
CLKOUT_PCIE_P9	U3
CNV_WR_CLKN	BD4
CNV_WR_CLKP	BE3
CNV_WR_D0N	BB3
CNV_WR_D0P	BB4
CNV_WR_D1N	BA3
CNV_WR_D1P	BA2
CNV_WT_CLKN	BC5
CNV_WT_CLKP	BB6
CNV_WT_D0N	BE6
CNV_WT_D0P	BD7
CNV_WT_D1N	BG6
CNV_WT_D1P	BF6
CNV_WT_RCOMP	BA1
CPU_TRST#	AM4
CPUPWRGD	AE3
DCPRTC	BF47
DCPRTC	BG47
DMI0_RXN	K34
DMI0_RXP	J35
DMI0_TXN	C33
DMI0_TXP	B33
DMI1_RXN	G33
DMI1_RXP	F34
DMI1_TXN	C32
DMI1_TXP	B32
DMI2_RXN	K32
DMI2_RXP	J32
DMI2_TXN	C31
DMI2_TXP	B31
DMI3_RXN	G30
DMI3_RXP	F30
DMI3_TXN	C29
DMI3_TXP	B29
DRAM_RESET#	BB46

Table 11-1. PCH BGA Ballout (Sheet 3 of 24)

Ball Name	Ball #
DSW_PWROK	AW41
GPD0 / BATLOW#	BF44
GPD1 / ACPRESENT	BG42
GPD10 / SLP_S5#	BC42
GPD11 / LANPHYPC	BF41
GPD2 / LAN_WAKE#	BG44
GPD3 / PWRBTN#	BE46
GPD4 / SLP_S3#	BF42
GPD5 / SLP_S4#	BE42
GPD6 / SLP_A#	BE40
GPD7	BE41
GPD8 / SUSCLK	BE45
GPD9 / SLP_WLAN#	BD42
GPP_A8 / CLKRUN#	AV32
GPP_A0/RCIN#/ ESPI_ALERT1#	BE39
GPP_A1/LAD0/ ESPI_IO0	BB39
GPP_A10 / CLKOUT_LPC1	BB34
GPP_A11 / PME# / SD_VDD2_PWR_EN#	BE36
GPP_A12 / BM_BUSY# / ISH_GP6 / SX_EXIT_HOLDOFF#	BF36
GPP_A13 / SUSWARN# / SUSPWRDNACK	BC37
GPP_A14 / SUS_STAT# / ESPI_RESET#	BF38
GPP_A15 / SUSACK#	BE35
GPP_A16 / CLKOUT_48	BE33
GPP_A17/ SD_VDD1_PWR_EN# / ISH_GP7	BD38
GPP_A18 / ISH_GP0	BF35
GPP_A19 / ISH_GP1	BD34
GPP_A2 / LAD1 / ESPI_IO1	AW37
GPP_A20 / ISH_GP2	BE34
GPP_A21 / ISH_GP3	BA33
GPP_A22 / ISH_GP4	AW32
GPP_A23 / ISH_GP5	AV34
GPP_A3/LAD2/ESPI_IO2	AV37
GPP_A4/LAD3/ ESPI_IO3	BA38
GPP_A5 / LFRAME# / ESPI_CS0#	BE38
GPP_A6 / SERIRQ / ESPI_CS1#	AW35


Table 11-1. PCH BGA Ballout (Sheet 4 of 24)

Ball Name	Ball #
GPP_A7 / PIRQA# / ESPI_ALERT0#	BA36
GPP_A9 / CLKOUT_LPC0 / ESPI_CLK	BB36
GPP_B0 / GSPI0_CS1#	BE29
GPP_B1 / GSPI1_CS1# / TIME_SYNC1	BF33
GPP_B10 / SRCCLKREQ5#	AN29
GPP_B11 / I2S_MCLK	AP29
GPP_B12 / SLP_S0#	BC28
GPP_B13 / PLTRST#	AV29
GPP_B14 / SPKR	AW29
GPP_B15 / GSPI0_CS0#	BB26
GPP_B16 / GSPI0_CLK	BF29
GPP_B17 / GSPI0_MISO	BD29
GPP_B18 / GSPI0_MOSI	BE30
GPP_B19 / GSPI1_CS0#	AW26
GPP_B2 / VRALERT#	BE32
GPP_B20 / GSPI1_CLK	AU26
GPP_B21 / GSPI1_MISO	BD30
GPP_B22 / GSPI1_MOSI	BA26
GPP_B23 / SML1ALERT# / PCHHOT#	BD33
GPP_B3 / CPU_GP2	BF32
GPP_B4 / CPU_GP3	BC33
GPP_B5 / SRCCLKREQ0#	BF31
GPP_B6 / SRCCLKREQ1#	BE31
GPP_B7 / SRCCLKREQ2#	AR32
GPP_B8 / SRCCLKREQ3#	BB30
GPP_B9 / SRCCLKREQ4#	BA30
GPP_C0 / SMBCLK	BE26
GPP_C1 / SMBDATA	BF26
GPP_C10 / UART0_RTS#	BA24
GPP_C11 / UART0_CTS#	AP24
GPP_C12 / UART1_RXD / ISH_UART1_RXD	AU24
GPP_C13 / UART1_TXD / ISH_UART1_TXD	AP21
GPP_C14 / UART1_RTS# / ISH_UART1_RTS#	AW24
GPP_C15 / UART1_CTS# / ISH_UART1_CTS#	BD21
GPP_C16 / I2C0_SDA	BF23
GPP_C17 / I2C0_SCL	BC22

Table 11-1. PCH BGA Ballout (Sheet 5 of 24)

Ball Name	Ball #
GPP_C18 / I2C1_SDA	BF21
GPP_C19 / I2C1_SCL	BE21
GPP_C2 / SMBALERT#	BE25
GPP_C20 / UART2_RXD	BD20
GPP_C21 / UART2_TXD	BE20
GPP_C22 / UART2_RTS#	AW21
GPP_C23 / UART2_CTS#	AV21
GPP_C3 / SML0CLK	BF25
GPP_C4 / SML0DATA	BE24
GPP_C5 / SML0ALERT#	BF24
GPP_C6 / SML1CLK	BF27
GPP_C7 / SML1DATA	BE27
GPP_C8 / UART0_RXD	BE23
GPP_C9 / UART0_TXD	BB24
GPP_D0 / SPI1_CS# / SBK0 / BK0	BF19
GPP_D1 / SPI1_CLK / SBK1 / BK1	BE19
GPP_D10 / ISH_SPI_CLK / GSPI2_CLK	BB20
GPP_D11 / ISH_SPI_MISO / GP_BSSB_CLK / GSPI2_MISO	BB16
GPP_D12 / ISH_SPI_MOSI / GP_BSSB_DI / GSPI2_MOSI	AN18
GPP_D13 / ISH_UART0_RXD / I2C2_SDA	BE17
GPP_D14 / ISH_UART0_TXD / I2C2_SCL	BF17
GPP_D15 / ISH_UART0_RTS# / GSPI2_CS1# / CNV_WFEN	AR18
GPP_D16 / ISH_UART0_CTS# / CNV_WCEN	BF14
GPP_D17 / DMIC_CLK1 / SNDW3_CLK	AW15
GPP_D18 / DMIC_DATA1 / SNDW3_DATA	AV16
GPP_D19 / DMIC_CLK0 / SNDW4_CLK	BD16
GPP_D2 / SPI1_MISO / SBK2 / BK2	BE18
GPP_D20 / DMIC_DATA0 / SNDW4_DATA	BF15
GPP_D21 / SPI1_IO2	BD17
GPP_D22 / SPI1_IO3	BC17

Table 11-1. PCH BGA Ballout (Sheet 6 of 24)

Ball Name	Ball #
GPP_D23 / ISH_I2C2_SCL / I2C3_SCL	BE14
GPP_D3 / SPI1_MOSI / SBK3 / BK3	BF18
GPP_D4 / ISH_I2C2_SDA / I2C3_SDA / SBK4 / BK4	BE15
GPP_D5 / I2S2_SFRM / CNV_RF_RESET#	BE16
GPP_D6 / I2S2_TXD / MODEM_CLKREQ	BA17
GPP_D7 / I2S2_RXD	AW18
GPP_D8 / I2S2_SCLK	AV18
GPP_D9 / ISH_SPI_CS# / GSPI2_CS0#	BA20
GPP_E0 / SATAXPCIE0 / SATAGP0	AH41
GPP_E1 / SATAXPCIE1 / SATAGP1	AJ43
GPP_E10 / USB2_OC1#	AL40
GPP_E11 / USB2_OC2#	AJ44
GPP_E12 / USB2_OC3#	AL41
GPP_E2 / SATAXPCIE2 / SATAGP2	AK47
GPP_E3 / CPU_GP0	AL47
GPP_E4 / SATA_DEVSLP0	AL48
GPP_E5 / SATA_DEVSLP1	AH35
GPP_E6 / SATA_DEVSLP2	AH40
GPP_E7 / CPU_GP1	AM45
GPP_E8 / SATALED#	AK48
GPP_E9 / USB2_OC0#	AH36
GPP_F0 / SATAXPCIE3 / SATAGP3	AN47
GPP_F1 / SATAXPCIE4 / SATAGP4	AM46
GPP_F10 / SATA_SCLOCK	AR42
GPP_F11 / SATA_SLOAD	AR48
GPP_F12 / SATA_SDATAOUT1	AU46
GPP_F13 / SATA_SDATAOUT0	AU47
GPP_F14 / PS_ON#	AP41
GPP_F15 / USB2_OC4#	AV47
GPP_F16 / USB2_OC5#	AR35
GPP_F17 / USB2_OC6#	AR37
GPP_F18 / USB2_OC7#	AV43
GPP_F19 / eDP_VDDEN	AV44
GPP_F2 / SATAXPCIE5 / SATAGP5	AM43



Table 11-1. PCH BGA Ballout (Sheet 7 of 24)

Ball Name	Ball #
GPP_F20 / eDP_BKLTEN	AV46
GPP_F21 / eDP_BKLTCTL	AU48
GPP_F22 / DDPF_CTRLCLK	AT49
GPP_F23 / DDPF_CTRLDATA	AN40
GPP_F3 / SATAXPICIE6 / SATAGP6	AM47
GPP_F4 / SATAXPICIE7 / SATAGP7	AM48
GPP_F5 / SATA_DEVSLP3	AP48
GPP_F6 / SATA_DEVSLP4	AR47
GPP_F7 / SATA_DEVSLP5	AN46
GPP_F8 / SATA_DEVSLP6	AN37
GPP_F9 / SATA_DEVSLP7	AP47
GPP_G0 / SD_CMD	AW13
GPP_G1 / SD_DATA0	BE9
GPP_G2 / SD_DATA1	BF8
GPP_G3 / SD_DATA2	BF9
GPP_G4 / SD_DATA3	BG8
GPP_G5 / SD_CD#	BE8
GPP_G6 / SD_CLK	BD8
GPP_G7 / SD_WP	AV13
GPP_H0 / SRCCLKREQ6#	AE47
GPP_H1 / SRCCLKREQ7#	AC48
GPP_H10 / SML2CLK	AE48
GPP_H11 / SML2DATA	AD47
GPP_H12 / SML2ALERT#	AB47
GPP_H13 / SML3CLK	AF47
GPP_H14 / SML3DATA	AD48
GPP_H15 / SML3ALERT#	AC47
GPP_H16 / SML4CLK	AE43
GPP_H17 / SML4DATA	AJ46
GPP_H18 / SML4ALERT#	AE44
GPP_H19 / ISH_I2C0_SDA	AH46
GPP_H2 / SRCCLKREQ8#	AE41
GPP_H20 / ISH_I2C0_SCL	AG45
GPP_H21 / ISH_I2C1_SDA	AH48
GPP_H22 / ISH_I2C1_SCL	AH47
GPP_H23 / TIME_SYNC0	AJ47
GPP_H3 / SRCCLKREQ9#	AF48

Table 11-1. PCH BGA Ballout (Sheet 8 of 24)

Ball Name	Ball #
GPP_H4 / SRCCLKREQ10#	AC41
GPP_H5 / SRCCLKREQ11#	AC39
GPP_H6 / SRCCLKREQ12#	AE39
GPP_H7 / SRCCLKREQ13#	AB48
GPP_H8 / SRCCLKREQ14#	AC44
GPP_H9 / SRCCLKREQ15#	AC43
GPP_I0 / DDPB_HPD0 / DISP_MISC0	AT6
GPP_I1 / DDPC_HPD1 / DISP_MISC1	AN10
GPP_I10 / DDPD_CTRLDATA	AR3
GPP_I11 / M2_SKT2_CFG0	AP3
GPP_I12 / M2_SKT2_CFG1	AP2
GPP_I13 / M2_SKT2_CFG2	AN4
GPP_I14 / M2_SKT2_CFG3	AM7
GPP_I2 / DDPD_HPD2 / DISP_MISC2	AP9
GPP_I3 / DDPF_HPD3 / DISP_MISC3	AL15
GPP_I4 / EDP_HPD / DISP_MISC4	AN6
GPP_I5 / DDPB_CTRLCLK	AL13
GPP_I6 / DDPB_CTRLDATA	AR8
GPP_I7 / DDPC_CTRLCLK	AN13
GPP_I8 / DDPC_CTRLDATA	AL10
GPP_I9 / DDPD_CTRLCLK	AL9
GPP_J2	AW3
GPP_J3	AT10
GPP_J4 / CNV_BRI_DT / UART0B_RTS#	AV4
GPP_J0 / CNV_PA_BLANKING	AV6
GPP_J1 / CPU_C10_GATE#	AY3
GPP_J10	AV7
GPP_J11 / A4WP_PRESENT	AR13
GPP_J5 / CNV_BRI_RSP / UART0B_RXD	AY2
GPP_J6 / CNV_RGI_DT / UART0B_TXD	BA4
GPP_J7 / CNV_RGI_RSP / UART0B_CTS#	AV3
GPP_J8 / CNV_MFUART2_RXD	AW2

Table 11-1. PCH BGA Ballout (Sheet 9 of 24)

Ball Name	Ball #
GPP_J9 / CNV_MFUART2_TXD	AU9
GPP_K0	L47
GPP_K1	L46
GPP_K10	V48
GPP_K11	W47
GPP_K12 / GSXDOUT	Y46
GPP_K13 / GSXSLOAD	Y48
GPP_K14 / GSXDIN	W46
GPP_K15 / GSXSRESET#	AA45
GPP_K16 / GSXCLK	Y47
GPP_K17 / ADR_COMPLETE	R47
GPP_K18 / NMI#	T47
GPP_K19 / SMI#	T48
GPP_K2	U48
GPP_K20	T46
GPP_K21	T45
GPP_K22 / IMGCLKOUT0	L48
GPP_K23 / IMGCLKOUT1	M45
GPP_K3	U47
GPP_K4	N48
GPP_K5	N47
GPP_K6	P47
GPP_K7	R46
GPP_K8	P48
GPP_K9	V47
GPPJ_RCOMP_1P8	BD1
GPPJ_RCOMP_1P8	BE1
GPPJ_RCOMP_1P8	BE2
HDA_BCLK / I2S0_SCLK	BD11
HDA_RST# / I2S1_SCLK	BE10
HDA_SDI0 / I2S0_RXD	BE11
HDA_SDI1 / I2S1_RXD	BF10
HDA_SDO / I2S0_TXD	BF12
HDA_SYNC / I2S0_SFRM	BG13
HDACPU_SCLK	AM3
HDACPU_SDI	AN3
HDACPU_SDO	AM2
I2S1_SFRM / SNDW2_CLK	BD12


Table 11-1. PCH BGA Ballout (Sheet 10 of 24)

Ball Name	Ball #
I2S1_TXD / SNDW2_DATA	BE12
INTRUDER#	BB44
ITP_PMODE	AL3
PCH_JTAG_TCK	AJ3
PCH_JTAG_TDI	AH2
PCH_JTAG_TDO	AH3
PCH_JTAG_TMS	AJ4
PCH_JTAGX	AH4
PCH_PWROK	AY42
PCIE_RCOMP_N	B12
PCIE_RCOMP_P	A13
PCIE1_RXN / USB31_7_RXN	G17
PCIE1_RXP / USB31_7_RXP	F16
PCIE1_TXN / USB31_7_TXN	A17
PCIE1_TXP / USB31_7_TXP	B17
PCIE10_RXN	K37
PCIE10_RXP	J37
PCIE10_TXN	C35
PCIE10_TXP	B35
PCIE11_RXN / SATA0A_RXN	G38
PCIE11_RXP / SATA0A_RXP	F39
PCIE11_TXN / SATA0A_TXN	B36
PCIE11_TXP / SATA0A_TXP	C36
PCIE12_RXN / SATA1A_RXN	H42
PCIE12_RXP / SATA1A_RXP	J41
PCIE12_TXN / SATA1A_TXN	D38
PCIE12_TXP / SATA1A_TXP	E37
PCIE13_RXN / SATA0B_RXN	C45
PCIE13_RXP / SATA0B_RXP	C46
PCIE13_TXN / SATA0B_TXN	B38
PCIE13_TXP / SATA0B_TXP	C38
PCIE14_RXN / SATA1B_RXN	D46
PCIE14_RXP / SATA1B_RXP	C47

Table 11-1. PCH BGA Ballout (Sheet 11 of 24)

Ball Name	Ball #
PCIE14_TXN / SATA1B_TXN	C39
PCIE14_TXP / SATA1B_TXP	D39
PCIE15_RXN / SATA2_RXN	F44
PCIE15_RXP / SATA2_RXP	E45
PCIE15_TXN / SATA2_TXN	B40
PCIE15_TXP / SATA2_TXP	C40
PCIE16_RXN / SATA3_RXN	L41
PCIE16_RXP / SATA3_RXP	M40
PCIE16_TXN / SATA3_TXN	B41
PCIE16_TXP / SATA3_TXP	C41
PCIE17_RXN / SATA4_RXN	K43
PCIE17_RXP / SATA4_RXP	K44
PCIE17_TXN / SATA4_TXN	A42
PCIE17_TXP / SATA4_TXP	B42
PCIE18_RXN / SATA5_RXN	P41
PCIE18_RXP / SATA5_RXP	R40
PCIE18_TXN / SATA5_TXN	C42
PCIE18_TXP / SATA5_TXP	D42
PCIE19_RXN / SATA6_RXN	M44
PCIE19_RXP / SATA6_RXP	N42
PCIE19_TXN / SATA6_TXN	C44
PCIE19_TXP / SATA6_TXP	D43
PCIE2_RXN / USB31_8_RXN	R21
PCIE2_RXP / USB31_8_RXP	P21
PCIE2_TXN / USB31_8_TXN	B18
PCIE2_TXP / USB31_8_TXP	C18
PCIE20_RXN / SATA7_RXN	R35
PCIE20_RXP / SATA7_RXP	R37
PCIE20_TXN / SATA7_TXN	A44
PCIE20_TXP / SATA7_TXP	B44
PCIE21_RXN	T43
PCIE21_RXP	R44
PCIE21_TXN	G47
PCIE21_TXP	F46
PCIE22_RXN	U40

Table 11-1. PCH BGA Ballout (Sheet 12 of 24)

Ball Name	Ball #
PCIE22_RXP	U41
PCIE22_TXN	H47
PCIE22_TXP	H48
PCIE23_RXN	W43
PCIE23_RXP	W44
PCIE23_TXN	G49
PCIE23_TXP	G48
PCIE24_RXN	Y40
PCIE24_RXP	Y41
PCIE24_TXN	G46
PCIE24_TXP	G45
PCIE3_RXN / USB31_9_RXN	K18
PCIE3_RXP / USB31_9_RXP	J18
PCIE3_TXN / USB31_9_TXN	B19
PCIE3_TXP / USB31_9_TXP	C19
PCIE4_RXN / USB31_10_RXN	N18
PCIE4_RXP / USB31_10_RXP	R18
PCIE4_TXN / USB31_10_TXN	D20
PCIE4_TXP / USB31_10_TXP	C20
PCIE5_RXN	F20
PCIE5_RXP	G20
PCIE5_TXN	B21
PCIE5_TXP	A22
PCIE6_RXN	K21
PCIE6_RXP	J21
PCIE6_TXN	D21
PCIE6_TXP	C21
PCIE7_RXN	L24
PCIE7_RXP	J24
PCIE7_TXN	C23
PCIE7_TXP	B23
PCIE8_RXN	F24
PCIE8_RXP	G24
PCIE8_TXN	B24
PCIE8_TXP	C24
PCIE9_RXN	G36



Table 11-1. PCH BGA Ballout (Sheet 13 of 24)

Ball Name	Ball #
PCIE9_RXP	F36
PCIE9_TXN	C34
PCIE9_TXP	D34
PECI	AF2
PLTRST_CPU#	AG5
PM_DOWN	AE2
PM_SYNC	AF3
PRDY#	AM5
PREQ#	AL2
RSMRST#	BA47
RSVD	BC1
RSVD	AH14
RSVD	AH15
RSVD	R32
RSVD	N32
RSVD	U35
RSVD	U37
RSVD	R15
RSVD	R13
RSVD	U13
RSVD	Y35
RSVD	Y36
RSVD	Y15
RSVD	Y14
RSVD	M29
RSVD	K29
RSVD	E28
RSVD	D29
RSVD	M26
RSVD	L26
RSVD	C27
RSVD	B27
RSVD	G26
RSVD	F26
RSVD	B26
RSVD	C26
RSVD	R24
RSVD	P24
RSVD	B25

Table 11-1. PCH BGA Ballout (Sheet 14 of 24)

Ball Name	Ball #
RSVD	A25
RTCRST#	BE47
RTCX1	BA49
RTCX2	BA48
SD_1P8_RCOMP	BE5
SD_3P3_RCOMP	BE4
SLP_LAN#	BF40
SLP_SUS#	BD39
SPI0_CLK	AW47
SPI0_CS0#	AY47
SPI0_CS1#	AW48
SPI0_CS2#	AT40
SPI0_IO2	AY48
SPI0_IO3	BA46
SPI0_MISO	BA45
SPI0_MOSI	AU41
SRTCST#	BD46
SYS_PWROK	AU3
SYS_RESET#	AU2
THRMTrip#	AD3
TP	AL35
TP	AN35
TRIGGER_IN	AK2
TRIGGER_OUT	AK3
USB2_COMP	F4
USB2_ID	G3
USB2_VBUSSENSE	F3
USB2N_1	J3
USB2N_10	H3
USB2N_11	R10
USB2N_12	G1
USB2N_13	N3
USB2N_14	E5
USB2N_2	N13
USB2N_3	K4
USB2N_4	M10
USB2N_5	M1
USB2N_6	K7
USB2N_7	L4

Table 11-1. PCH BGA Ballout (Sheet 15 of 24)

Ball Name	Ball #
USB2N_8	G4
USB2N_9	M6
USB2P_1	J2
USB2P_10	H2
USB2P_11	P9
USB2P_12	G2
USB2P_13	N2
USB2P_14	F6
USB2P_2	N15
USB2P_3	K3
USB2P_4	L9
USB2P_5	L2
USB2P_6	K6
USB2P_7	L3
USB2P_8	G5
USB2P_9	N8
USB31_1_RXN	D11
USB31_1_RXP	C11
USB31_1_TXN	F9
USB31_1_TXP	F7
USB31_2_RXN	B9
USB31_2_RXP	C9
USB31_2_TXN	C3
USB31_2_TXP	D4
USB31_3_RXN	B10
USB31_3_RXP	C10
USB31_3_TXN	F11
USB31_3_TXP	G12
USB31_4_RXN	K16
USB31_4_RXP	J15
USB31_4_TXN	B14
USB31_4_TXP	C14
USB31_5_RXN	J13
USB31_5_RXP	K13
USB31_5_TXN	C15
USB31_5_TXP	B15
USB31_6_RXN	G14
USB31_6_RXP	F14
USB31_6_TXN	C17


Table 11-1. PCH BGA Ballout (Sheet 16 of 24)

Ball Name	Ball #
USB31_6_TXP	C16
VCCA_BCLK_1P05	V19
VCCA_SRC_1P05	W19
VCCA_SRC_1P05	W20
VCCA_XTAL_1P05	P2
VCCA_XTAL_1P05	P3
VCCAMPHYPLL_1P05	C49
VCCAMPHYPLL_1P05	D49
VCCAMPHYPLL_1P05	E49
VCCAPLL_1P05	B1
VCCAPLL_1P05	B2
VCCAPLL_1P05	B3
VCCAPLL_1P05	C1
VCCAPLL_1P05	C2
VCCDPHY_1P24	AJ22
VCCDPHY_1P24	AJ23
VCCDPHY_1P24	BG5
VCCDSW_1P05	BG45
VCCDSW_1P05	BG46
VCCDSW_3P3	BE48
VCCDSW_3P3	BE49
VCCDUSB_1P05	W22
VCCDUSB_1P05	W23
VCCHDA	BB14
VCCMPHY_SENSE	K47
VCCPGPPA	AN32
VCCPGPPBC	AN26
VCCPGPPBC	AP26
VCCPGPPD	AN24
VCCPGPPEF	AE35
VCCPGPPEF	AE36
VCCPGPPG_3P3	AN21
VCCPGPPHK	AC35
VCCPGPPHK	AC36
VCCPHVLDO_1P8	AF19
VCCPHVLDO_1P8	AF20
VCCPRIM_1P05	D1
VCCPRIM_1P05	E1
VCCPRIM_1P05	AF31

Table 11-1. PCH BGA Ballout (Sheet 17 of 24)

Ball Name	Ball #
VCCPRIM_1P05	AG31
VCCDPHY_1P24	AK22
VCCDPHY_1P24	AK23
VCCPRIM_1P05	U26
VCCPRIM_1P05	U29
VCCPRIM_1P05	V25
VCCPRIM_1P05	V27
VCCPRIM_1P05	V28
VCCPRIM_1P05	V30
VCCPRIM_1P05	V31
VCCPRIM_1P05	AA22
VCCPRIM_1P05	AA23
VCCPRIM_1P05	AB20
VCCPRIM_1P05	AB22
VCCPRIM_1P05	AB23
VCCPRIM_1P05	AB27
VCCPRIM_1P05	AB28
VCCPRIM_1P05	AB30
VCCPRIM_1P05	AD20
VCCPRIM_1P05	AD23
VCCPRIM_1P05	AD27
VCCPRIM_1P05	AD28
VCCPRIM_1P05	AD30
VCCPRIM_1P05	AF23
VCCPRIM_1P05	AF27
VCCPRIM_1P05	AF30
VCCPRIM_1P05	AE17
VCCPRIM_1P05	AD31
VCCPRIM_1P8	AG19
VCCPRIM_1P8	AG20
VCCPRIM_1P8	AN15
VCCPRIM_1P8	AR15
VCCPRIM_1P8	BB11
VCCPRIM_3P3	AT44
VCCPRIM_3P3	AW9
VCCPRIM_3P3	AY8
VCCPRIM_3P3	BB7
VCCPRIM_3P3	V23

Table 11-1. PCH BGA Ballout (Sheet 18 of 24)

Ball Name	Ball #
VCCPRIM_MPHY_1P05	W31
VCCRTC	BC49
VCCRTC	BD49
VCCSPI	AN44
VSS	AL37
VSS	A2
VSS	A28
VSS	A3
VSS	A33
VSS	A37
VSS	A4
VSS	A45
VSS	A46
VSS	A47
VSS	A48
VSS	A5
VSS	A8
VSS	AA19
VSS	AA20
VSS	AA25
VSS	AA27
VSS	AA28
VSS	AA30
VSS	AA31
VSS	AA49
VSS	AA5
VSS	AB19
VSS	AB25
VSS	AB31
VSS	AC12
VSS	AC17
VSS	AC33
VSS	AC38
VSS	AC4
VSS	AC46
VSS	AD1
VSS	AD19
VSS	AD2



Table 11-1. PCH BGA Ballout (Sheet 19 of 24)

Ball Name	Ball #
VSS	AD22
VSS	AD25
VSS	AD49
VSS	AE12
VSS	AE33
VSS	AE38
VSS	AE4
VSS	AE46
VSS	AF22
VSS	AF25
VSS	AF28
VSS	AG1
VSS	AG22
VSS	AG23
VSS	AG25
VSS	AG27
VSS	AG28
VSS	AG30
VSS	AG49
VSS	AH12
VSS	AH17
VSS	AH33
VSS	AH38
VSS	AJ19
VSS	AJ20
VSS	AJ25
VSS	AJ27
VSS	AJ28
VSS	AJ30
VSS	AJ31
VSS	AK19
VSS	AK20
VSS	AK25
VSS	AK27
VSS	AK28
VSS	AK30
VSS	AK31
VSS	AK4
VSS	AK46

Table 11-1. PCH BGA Ballout (Sheet 20 of 24)

Ball Name	Ball #
VSS	AL12
VSS	AL17
VSS	AL21
VSS	AL24
VSS	AL26
VSS	AL29
VSS	AL33
VSS	AL38
VSS	AM1
VSS	AM18
VSS	AM32
VSS	AM49
VSS	AN12
VSS	AN16
VSS	AN34
VSS	AN38
VSS	AP4
VSS	AP46
VSS	AR12
VSS	AR16
VSS	AR34
VSS	AR38
VSS	AT1
VSS	AT16
VSS	AT18
VSS	AT21
VSS	AT24
VSS	AT26
VSS	AT29
VSS	AT32
VSS	AT34
VSS	AT45
VSS	AV11
VSS	AV39
VSS	AW10
VSS	AW4
VSS	AW40
VSS	AW46
VSS	B47

Table 11-1. PCH BGA Ballout (Sheet 21 of 24)

Ball Name	Ball #
VSS	B48
VSS	B49
VSS	BA12
VSS	BA14
VSS	BA44
VSS	BA5
VSS	BA6
VSS	BB41
VSS	BB43
VSS	BB9
VSS	BC10
VSS	BC13
VSS	BC15
VSS	BC19
VSS	BC24
VSS	BC26
VSS	BC31
VSS	BC35
VSS	BC40
VSS	BC45
VSS	BC8
VSS	BD43
VSS	BE44
VSS	BF1
VSS	BF2
VSS	BF3
VSS	BF48
VSS	BF49
VSS	BG17
VSS	BG2
VSS	BG22
VSS	BG25
VSS	BG28
VSS	BG3
VSS	BG33
VSS	BG37
VSS	BG4
VSS	BG48
VSS	C12



Table 11-1. PCH BGA Ballout (Sheet 22 of 24)

Ball Name	Ball #
VSS	C25
VSS	C30
VSS	C4
VSS	C48
VSS	C5
VSS	D12
VSS	D16
VSS	D17
VSS	D30
VSS	D33
VSS	D8
VSS	E10
VSS	E13
VSS	E15
VSS	E17
VSS	E19
VSS	E22
VSS	E24
VSS	E26
VSS	E31
VSS	E33
VSS	E35
VSS	E40
VSS	E42
VSS	E8
VSS	F41
VSS	F43
VSS	F47
VSS	G44
VSS	G6
VSS	H8
VSS	J10
VSS	J26
VSS	J29
VSS	J4
VSS	J40
VSS	J46
VSS	J47

Table 11-1. PCH BGA Ballout (Sheet 23 of 24)

Ball Name	Ball #
VSS	J48
VSS	J9
VSS	K11
VSS	K39
VSS	M16
VSS	M18
VSS	M21
VSS	M24
VSS	M32
VSS	M34
VSS	M49
VSS	M5
VSS	N12
VSS	N16
VSS	N34
VSS	N35
VSS	N37
VSS	N38
VSS	P26
VSS	P29
VSS	P4
VSS	P46
VSS	R12
VSS	R16
VSS	R26
VSS	R29
VSS	R3
VSS	R34
VSS	R38
VSS	R4
VSS	T17
VSS	T18
VSS	T32
VSS	T4
VSS	T49
VSS	T5
VSS	T7
VSS	U12

Table 11-1. PCH BGA Ballout (Sheet 24 of 24)

Ball Name	Ball #
VSS	U15
VSS	U17
VSS	U21
VSS	U24
VSS	U33
VSS	U38
VSS	V20
VSS	V22
VSS	V4
VSS	V46
VSS	W25
VSS	W27
VSS	W28
VSS	W30
VSS	Y10
VSS	Y12
VSS	Y17
VSS	Y33
VSS	Y38
VSS	Y9
VSSMPHY_SENSE	K46
WAKE#	BB47
XCLK_BIASREF	T3
XTAL_IN	U10
XTAL_OUT	U9

§ §

12 8254 Timers

12.1 Overview

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 1.193 MHz periodic timer tick is generated off the 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).

12.1.1 Timer Programming

The counter/timers are programmed in the following fashion:

1. Write a control word to select a counter.
2. Write an initial count for that counter.
3. Load the least and/or most significant bytes (as required by Control Word Bits 5, 4) of the 16-bit counter.
4. Repeat with other counters.

Only two conventions need to be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant byte only, most significant byte only, or least significant byte, and then most significant byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read/write 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.

Table 12-1 lists the six operating modes for the interval counters.

Table 12-1. 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.

12.1.2 Reading from Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters—a simple read operation, counter Latch command, and the Read-Back command. Each 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.

12.1.2.1 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.



12.1.2.2 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.

12.1.2.3 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.





13 Audio, Voice, and Speech

13.1 Acronyms

Acronyms	Description
DMIC	Digital Microphone. PDM based MEMs microphone modules
DSP	Digital Signal Processor. In AVS specifically an DSP to process audio data.
I ² S	Inter IC Sound. A serial bus using PCM.
LDO	Low Drop Out. A type of voltage regulator that can regulate output voltage when input voltage is close to output voltage.
MEMs	Micro electrical mechanical Systems. For AVS devices such as Digital MEMs Microphones.
PCM	Pulse Code Modulation. Modulation with amplitude coded into stream.
PDM	Pulse Density Modulation. Modulation with amplitude coded by pulse density.
SoC	System On Chip.
VAD	Voice Activity Detector.
VOIP	Voice Over Internet Protocol

13.2 AVS Subsystem Overview

This section will cover some of the details of the separate subsystem parts and how they interact starting with the Intel[®] HD Audio Controller.

The Intel HD Audio controller basically consists of a set of DMA engines that are used to move samples of digitally encoded data between system memory and an internal/external codec(s), with the optional Intel[®] Smart Sound Technology (DSPs) enhancements. The controller communicates with the internal/external codec(s) over the Audio Link. The Audio link is a collection of Intel High Definition Audio serial link, Intel[®] Display Audio serial link, or other DSP I/O peripheral for proprietary interfaces (example: I²S, MIPI* SoundWire*, DMIC).

The Intel HD Audio controller implements a number of output DMA engines and input DMA engines. The Output DMA engines move digital data from system memory to a DAC in a codec. The Input DMA engines move digital data from the ADC in the codec to system memory. The audio link supports a number of internal/external codecs by implementing Serial Data Input signals, one dedicated to each of the supported codecs. Audio software renders outbound, and processes inbound data to/from buffers in memory. The Output DMA engines fetch the digital data from memory and reformat it based on the programmed sample rate, bits/sample and number of channels. The data from the Output DMA engines is then combined and serially sent to the codec(s) over the audio link. The Input DMA engines receive data from the codec(s) over the audio link and format the data based on the programmable attributes for that stream. The data is then written to memory in the predefined format for software to process. Each DMA engine moves one "stream" of data. A single codec can accept or generate multiple "streams" of data, one for each ADC or DAC in the codec. Multiple codecs can accept the same output "stream" processed by a single DMA engine.



The audio subsystem contains serial audio interfaces (I²S, Intel Display Audio Interface), PDM digital microphone interfaces (DMIC), multi-context DMA controllers for transferring data between audio interfaces and memories, on-die SRAMs for storing data and code and serving as a shared L2 memory, and four processor cores (DSPs) based on Cadence* Tensilica* HiFi3/LX6 cores.

The Intel Smart Sound Technology subsystem contains four DSPs are located in their own power domains together with their L1 cache controllers and cache memories. The DSP domains may operate up to 400 MHz. One of the DSPs can be enabled in the D0ix state of the overall audio subsystem, the others are powered down and can be made operational only in D0 state.

The power gating and LDOs are controlled by the Local Power Sequencer (LPS) assisted by the DSP firmware. This approach allows for flexible and robust fine grain power management. The LPS communicates with the Power Management Controller (PMC) to request power supplies.

13.3 Signal Description

Table 13-1. Signal Descriptions (Sheet 1 of 3)

Name	Type	Description
Intel High Definition Audio Signals		
HDA_RST# / I2S1_SCLK / SNDW1_CLK	O	Intel HD Audio Reset: Master H/W reset to internal/external codecs.
HDA_SYNC / I2S0_SFRM	O	Intel HD Audio Sync: 48-KHz fixed rate frame sync to the codecs. Also used to encode the stream number.
HDA_BCLK / I2S0_SCLK	O	Intel HD Audio Bit Clock: Up to 24 MHz serial data clock generated by the Intel HD Audio controller.
HDA_SDO / I2S0_TXD	O	Intel HD Audio Serial Data Out: Serial TDM data output to the codecs. The serial output is double-pumped for a bit rate of up to 48 Mb/s.
HDA_SDI0 / I2S0_RXD	I/O	Intel HD Audio Serial Data In 0: Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated Pull-down resistors, which are enabled while the primary well is powered.
HDA_SDI1 / I2S1_RXD / SNDW1_DATA	I/O	Intel HD Audio Serial Data In 1: Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated Pull-down resistors, which are enabled while the primary well is powered.
Intel Display Audio Interface		
HDACPU_SCLK	O	Display Audio Bit Clock: Serial data clock generated by the Intel HD Audio controller. PCH supports data rate of up to 96 Mb/s.
HDACPU_SDO	O	Display Audio Serial Data Out: Serial TDM data output to the codec. PCH supports data rate of up to 96 Mb/s.
HDACPU_SDI	I	Display Audio Serial Data In: Serial TDM data input from the codec. PCH supports data rate of up to 96 Mb/s.
I²S/PCM Interface		
I2S0_SCLK / HDA_BCLK	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).
I2S1_SCLK / HDA_RST / SNDW1_CLK	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).



Table 13-1. Signal Descriptions (Sheet 2 of 3)

Name	Type	Description
I2S2_SCLK / GPP_D8	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).
I2S0_SFRM / HDA_SYNC	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).
I2S1_SFRM / SNDW2_CLK	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).
I2S2_SFRM / GPP_D5 CNV_RF_RESET#	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).
I2S0_TXD / HDA_SDO	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.
I2S1_TXD / SNDW2_DATA	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.
I2S2_TXD / GPP_D6 / 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.
I2S0_RXD / HDA_SDI0	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.
I2S1_RXD / HDA_SDI1	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.
I2S2_RXD / GPP_D7	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.
I2SBMCLK / GPP_B11	O	I²S/PCM Master reference clock: This signal is the master reference clock that connects to an audio codec.
DMIC Interface		
DMIC_CLK0 / GPP_D19 / SNDW4_CLK	O	Digital Mic Clock: Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz.
DMIC_CLK / GPP_D17 / SNDW3_CLK	O	Digital Mic Clock: Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz.
DMIC_DATA0 / GPP_D20 / SNDW4_DATA	I	Digital Mic Data: Serial data input from the digital mic.
DMIC_DATA1 / GPP_D18 / SNDW3_DATA	I	Digital Mic Data: Serial data input from the digital mic.
SoundWire Interface		
SNDW1_CLK / HDA_RST / I2S1_SCLK	O	SoundWire Clock: Serial data clock to external peripheral devices.
SNDW2_CLK / I2S1_SFRM /	O	SoundWire Clock: Serial data clock to external peripheral devices.
SNDW3_CLK / GPPC_D17 / DMIC_CLK1	O	SoundWire Clock: Serial data clock to external peripheral devices.
SNDW4_CLK / GPPC_D19 / DMIC_CLK0	O	SoundWire Clock: Serial data clock to external peripheral devices.
SNDW1_DATA / HDA_SDI1 / I2S1_RXD	I/O	SoundWire Data: Serial data input from external peripheral devices.



Table 13-1. Signal Descriptions (Sheet 3 of 3)

Name	Type	Description
SNDW2_DATA /I2S1_TXD	I/O	SoundWire Data: Serial data input from external peripheral devices.
SNDW3_DATA / GPPC_D18/ DMIC_DATA1	I/O	SoundWire Data: Serial data input from external peripheral devices.
SNDW4_DATA / GPPC_D20/ DMIC_DATA0	I/O	SoundWire Data: Serial data input from external peripheral devices.
Misc		
SPKR	0	Speaker Output: Used for connection to external speaker for POST sounds if not using the Intel HD_Audio embedded option.

13.4 Integrated Pull-Ups and Pull-Downs

Table 13-2. Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value
HDA_SYNC	Pull-down	14 kohm - 26 kohm
HDA_SDO	Pull-down	14 kohm - 26 kohm
HDA_SDI[1:0]	Pull-down	14 kohm - 26 kohm
HDACPU_SDO	Pull-down	14 kohm - 26 kohm
HDACPU_SDI	Pull-down	14 kohm - 26 kohm
I2S[2:0]_SFRM	Pull-down	14 kohm - 26 kohm
I2S[2:0]_RXD	Pull-down	14 kohm - 26 kohm
I2S[2:0]_SCLK	Pull-down	14 kohm - 26 kohm
DMIC_DATA[1:0]	Pull-down	14 kohm - 26 kohm
SNDW_DATA[3:0]	Pull-down	14 kohm - 26 kohm
SPKR	Pull-down	14 kohm - 26 kohm

13.5 I/O Signal Planes and States

Table 13-3. I/O Signal Planes and States (Sheet 1 of 2)

Signal Name	Power Plane	During Reset ²	Immediately After Reset ²	S3/S4/S5	Deep Sx
Intel High Definition Interface					
HDA_RST#	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
Intel Display Audio Interface					
HDACPU_BCLK	Primary	Driven Low	Driven Low	OFF	OFF

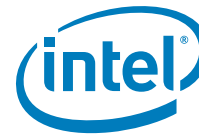


Table 13-3. I/O Signal Planes and States (Sheet 2 of 2)

Signal Name	Power Plane	During Reset ²	Immediately After Reset ²	S3/S4/S5	Deep Sx
HDACPU_SDO	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
HDACPU_SDI	Primary	Internal Pull-down	Internal Pull-down	OFF	OFF
I²S/PCM Interface					
I2S[2:0]_SCLK	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
I2S[2:0]_SFRM	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
I2S0_TXD	Primary	Internal Pull-down	Driven Low	Low then disabled (Refer Note)	OFF
I2S[2:0]_TXD	Primary	Driven Low	Driven Low	Driven Low	OFF
I2S[2:0]_RXD	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
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
SoundWire Interface					
SNDW_DATA[1:4]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SNDW_CLK[1:4]	Primary	Not Driven	Not Driven	Not Driven	OFF
Misc					
SPKR	Primary	Driven Low	Driven Low	Low then disabled (Refer note)	OFF
Notes:					
1. SPKR and I2S0_SFRM are also straps in which the 20 kohm pull-down only occurs during the sampling window and then the pull-ups are disabled.					
2. Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.					

13.6 AVS Feature Summary

13.6.1 Intel High Definition Audio Controller Capabilities

- PCI/PCI Express* controller
- Supports data transfers, descriptor fetches, and DMA position writes using VC0.
- Independent Bus Master logic for 16 general purpose streams: 7 input and 9 output
- Supports variable length stream slots
- Supports up to:
 - 16 streams (7 input and 9 output)
 - 16 channels per stream
 - 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)
 - Only if exposed as PCI Express device
- Support 1 ms of buffering with all DMA running with maximum bandwidth
- Support 10 ms of buffering with 1 output DMA and 1 input DMA running at 2 channels, 96 KHz, 16-bit audio

13.6.2 Audio DSP Capabilities

- DSP off load for low power audio rendering and recording
- Intel Wake on Voice supported in Connected Standby and D3/Sx
- High Performance DSP based on Cadence* Tensilica* LX6 HIFI3 DSP Cores operating up to 400 MHz
- Various DSP functions provided by DSP Core: MP3, AAC, 3rd Party IP Algorithms, etc.

13.6.3 Intel High Definition Audio Link Capabilities

- Two SDI signals to support two external codecs
- Drives variable frequency (6 MHz to 24 MHz) BCLK to support:
 - SDO double pumped up to 48 Mb/s
 - SDIs single pumped up to 24 Mb/s
- Provides cadence for 44.1 KHz-based sample rate output
- Supports 1.8v and 3.3v I/O voltages

13.6.4 Intel Display Audio Link Capabilities

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

13.6.5 DMIC Interface

- Two DMIC PDM interfaces with each port capable of supporting up to 4 digital MEMS microphones
- The audio clock output can operate up to 4.8 MHz
- Supports 1.8v and 3.3v I/O voltages



13.6.6 I²S/PCM Interface

- Three bi-directional I²S/PCM ports to support up to three I²S connections
- Support of serial bit rates of 19.2Mbps or 24Mbps
- Can support 3 modes: Slave Mode, Slave Mode with Locally Generated Master Clock, or Master Mode

13.6.7 SoundWire Interface

- 4 SoundWire Interfaces also referred to as ports for connection to platform peripherals
- Supports only 1.8v I/O voltage





14 Controller Link

14.1 Overview

The Controller Link interface is used to connect the Intel® CSME to a wireless LAN device supporting Intel® Active Management Technology. The Controller Link interface will transmit data at up to 60 Mbps with the clock frequency at 30 MHz.

14.2 Signal Description

Name	Type	Description
CL_DATA	I/O	Controller Link Data: Bi-directional data that connects to a Wireless LAN Device supporting Intel® Active Management Technology.
CL_CLK	I/O	Controller Link Clock: Bi-directional clock that connects to a Wireless LAN Device supporting Intel® Active Management Technology.
CL_RST#	O OD	Controller Link Reset: Controller Link reset that connects to a Wireless LAN Device supporting Intel® Active Management Technology.

14.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value (Ohm)	Notes
CL_DATA	Pull-up Pull-down	31.25 100	Refer Section 14.4
CL_CLK	Pull-up Pull-down	31.25 100	Refer Section 14.4

14.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ³	Immediately after Reset ³	S3/S4/S5	Deep Sx
CL_DATA	Primary	Refer Notes 1 and 2	Refer Notes	Internal Pull-down	Off
CL_CLK	Primary	Refer Notes 1 and 2	Refer Notes	Internal Pull-down	Off
CL_RST#	Primary	Driven Low	Driven High	Driven High	Off

Notes:

- The Controller Link clock and data buffers use internal Pull-up or Pull-down resistors to drive a logical 1 or 0.
- The terminated state is when the I/O buffer Pull-down is enabled.
- Reset reference for primary well pins is RSMRST#



14.5 External CL_RST# Pin Driven/Open-drain Mode Support

The WLAN has transitioned to 1.8 V for external CL_RST# pin, while PCH Controller Link I/O buffer still drives 3.3 V on this pin. This creates voltage in-compatibility issue. In order to support either 1.8 V or 3.3 V on the device CL_RST# pin, the PCH operates/controls the CL_RST# pin as dual modes, which is determined by a Soft-strap bit:

(1) Driven mode: To drive "1" on this pin, Controller Link turn-on the output enable and output=1 to drive 3.3 V on this pin. This mode can only be enabled with older version of WLAN which is 3.3 V tolerant.

(2) Open-drain mode: To drive "1", Controller Link turn-off the output-enable, and external (required) pull-up will pull the pin up to 1.8 V, which is compatible with WLAN voltage requirement.





15 Processor Sideband Signals

15.1 Acronyms

Acronyms	Description
PECI	Platform Environmental Control Interface

15.2 Overview

The sideband signals are used for the communication between the processor and PCH.

15.3 Signal Description

Name	Type	Description
CPUPWRGD	O	Signal to the processor to indicate its primary power is good.
THRMTRIP#	I	Signal from the processor to indicate that a thermal overheating has occurred.
PM_SYNC	O	Power Management Sync: State exchange from the PCH to the Processor
PM_DOWN	I	Power Management Sync: State exchange from the Processor to the PCH
PLTRST_CPU#	O	Platform reset to the Processor
PECI	I/O	Single-wire serial bus for accessing processor digital thermometer
GPP_E3 / CPU_GP0	I	Thermal management signal
GPP_E7 / CPU_GP1	I	Thermal management signal
GPP_B3 / CPU_GP2	I	Thermal management signal
GPP_B4 / CPU_GP3	I	Thermal management signal

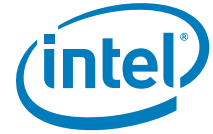
15.4 Integrated Pull-Ups and Pull-Downs

None

15.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset (Note 1)	Immediately after Reset (Note 1)	S3/S4/S5	Deep Sx
CPUPWRGD	Primary	Undriven	Driven High	Off	Off
THRMTRIP#	Primary	Undriven	Undriven	Off	Off
PM_SYNC	Primary	Driven Low	Driven Low	Off	Off
PM_DOWN	Primary	Undriven	Undriven	Off	Off
PLTRST_CPU#	Primary	Driven Low	Driven High	Off	Off
PECI	Primary	Undriven	Undriven	Off	Off
CPU_GP[3:0]	Primary	Undriven	Undriven	Undriven	Off

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



15.6 Functional Description

CPUPWRGD signal is output to the processor that indicates that the primary power is ramped up and stable. CPUPWRGD will be Undriven by the PCH (high Z) when RSMRST# is asserted and driven high after RSMRST# is De-asserted

If THRMTRIP# goes active, the processor is indicating an overheat condition, and the PCH will immediately transition to an S5 state. CPU_GP can be used from external sensors for the thermal management.

PM_SYNC is used to provide early warning to the processor that a global reset is in progress and that the memory contents should be saved and placed into self refresh.

PM_DOWN is input to PCH indicates the processor wake up event.

§ §



16 Digital Display Signals

16.1 Acronyms

Acronyms	Description
eDP*	embedded Display Port*

16.2 Signal Description

Display is divided between processor and PCH. The processor houses memory interface, display planes, pipes, and digital display interfaces/ports, while the PCH has DDC bus, Hot-Plug Detect, Panel Power and Backlight controls.

The DDC (Display Data Channel) bus is used for communication between the host system and display. A pair of DDC (DDC_CLK and DDC_DATA) signals exist on the PCH for each digital port on the processor. DDC is based on I²C protocol.

The Hot-Plug Detect (HPD) signal serves as an interrupt request for the sink device for DisplayPort* and HDMI*. It is a 3.3V tolerant signal pin on the PCH.

The Panel Power and Backlight controls are used to control power for an internal panel and drive the backlight.

Table 16-1. Digital Display Signals (Sheet 1 of 2)

Name	Type	Description
DDPB_HPDP0 / GPP_I0 / DISP_MISC0	I	Display Port B: HPD Hot-Plug Detect
DDPC_HPDP1 / GPP_I1 / DISP_MISC1	I	Display Port C: HPD Hot-Plug Detect
DDPD_HPDP2 / GPP_I2 / DISP_MISC2	I	Display Port D: HPD Hot-Plug Detect
DDPF_HPDP3 / GPP_I3 / DISP_MISC3	I	Display Port F: HPD Hot-Plug Detect
DDPB_CTRLCLK / GPP_I5	I/O	Display Port B: Control Clock.
DDPB_CTRLDATA / GPP_I6	I/O	Display Port B: Control Data.
DDPC_CTRLCLK / GPP_I7	I/O	Display Port C: Control Clock
DDPC_CTRLDATA / GPP_I8	I/O	Display Port C: Control Data
DDPD_CTRLCLK / GPP_I9	I/O	Display Port D: Control Clock
DDPD_CTRLDATA / GPP_I10	I/O	Display Port D: Control Data
DDPF_CTRLCLK / GPP_F22	I/O	Display Port F: Control Clock



Table 16-1. Digital Display Signals (Sheet 2 of 2)

Name	Type	Description
DDPF_CTRLDATA / GPP_F23	I/O	Display Port F: Control Data

16.3 Embedded DisplayPort* (eDP*) Backlight Control Signals

Name	Type	Description
eDP_VDDEN / GPP_F19	O	eDP Panel power Enable: Panel power control enable. This signal is used to control the VDC source of the panel logic.
eDP_BKLTEN / GPP_F20	O	eDP Backlight Enable: Panel backlight enable control for eDP. This signal is used to gate power into the backlight circuitry.
eDP_BKLTCTL / GPP_F21	O	eDP Panel Backlight Brightness control: Panel brightness control for eDP. This signal is used as the PWM Clock input signal
EDP_HPD / GPP_I4 / DISP_MISC4	I	eDP: Hot-Plug Detect
Note: eDP_VDDEN, eDP_BKLTEN, eDP_BKLTCTL can be left as no connect if eDP* is not used.		

16.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
DDPB_CTRLDATA / GPP_I6	Pull-down	15 kohm-40 kohm	Refer note below
DDPC_CTRLDATA / GPP_I7	Pull-down	15 kohm-40 kohm	Refer note below
DDPD_CTRLDATA / GPP_I10	Pull-down	15 kohm-40 kohm	Refer note below
DDPF_CTRLDATA / GPP_F23	Pull-down	15 kohm-40 kohm	Refer note below

Note: The internal pull-up/pull-down is only applied during the strap sampling window (PCH_PWROK) and is then disabled. Enabling can be done using a 2.2 kohm Pull-up resistor.

16.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ²	Immediately after Reset ²	S3/S4/S5	Deep Sx
DDPB_HPD0 / GPP_I0 / DISP_MISC0	Primary	Undriven	Undriven	Undriven	OFF
DDPC_HPD1 / GPP_I1 / DISP_MISC1	Primary	Undriven	Undriven	Undriven	OFF
DDPD_HPD2 / GPP_I2 / DISP_MISC2	Primary	Undriven	Undriven	Undriven	OFF



Signal Name	Power Plane	During Reset ²	Immediately after Reset ²	S3/S4/S5	Deep Sx
DDPF_HPD3 / GPP_I3 / DISP_MISC3	Primary	Undriven	Undriven	Undriven	OFF
DDPB_CTRLCLK / GPP_I5	Primary	Undriven	Undriven	Undriven	OFF
DDPB_CTRLDATA / GPP_I6	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
DDPC_CTRLCLK / GPP_I7	Primary	Undriven	Undriven	Undriven	OFF
DDPC_CTRLDATA / GPP_I8	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
DDPD_CTRLCLK / GPP_I9	Primary	Undriven	Undriven	Undriven	OFF
DDPD_CTRLDATA / GPP_I10	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
DDPF_CTRLCLK / GPP_F22	Primary	Undriven	Undriven	Undriven	OFF
DDPF_CTRLDATA / GPP_F23	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
eDP_VDDEN / GPP_F19	Primary	Driven Low	Driven Low	Driven Low	OFF
eDP_BKLTEN / GPP_F20	Primary	Driven Low	Driven Low	Driven Low	OFF
eDP_BKLTCTL / GPP_F21	Primary	Driven Low	Driven Low	Driven Low	OFF
EDP_HPD / GPP_I4 / DISP_MISC4	Primary	Undriven	Undriven	Undriven	OFF

§ §



17 Enhanced Serial Peripheral Interface (eSPI)

17.1 Acronyms

Acronyms	Description
EC	Embedded Controller
MAFCC	Master Attached Flash Channel Controller (MAFCC)
OOB	Out-of-Band
TAR	Turn-around cycle

17.2 Overview

The PCH provides the Enhanced Serial Peripheral Interface (eSPI) to support connection of an EC (typically used in mobile platform) or an SIO (typically used in desktop platform) to the platform.

The interface supports 1.8V only. Depending on PCH SKU, the interface can support either one device (via one chip select signal) or two devices (via two chip select signals). Refer to the PCH SKU section in Chapter 1 for more information. This interface is not shared and distinct from the SPI bus interface used for flash device and TPM.

Note: The PCH LPC and eSPI are mutually exclusive. A hardware strap is used to determine which interface is used on the platform.

17.3 Signal Description

Name	Type	Description
ESPI_CLK/ CLKOUT_LPC0/ GPP_A9	O	eSPI Clock: eSPI clock output from the PCH to slave device.
ESPI_IO0/ LAD0/GPP_A1	I/O	eSPI Data Signal 0: Bi-directional pin used to transfer data between the PCH and eSPI slave device.
ESPI_IO1/ LAD1/GPP_A2	I/O	eSPI Data Signal 1: Bi-directional pin used to transfer data between the PCH and eSPI slave device.
ESPI_IO2/ LAD2/GPP_A3	I/O	eSPI Data Signal 2: Bi-directional pin used to transfer data between the PCH and eSPI slave device.
ESPI_IO3/ LAD3/GPP_A4	I/O	eSPI Data Signal 3: Bi-directional pin used to transfer data between the PCH and eSPI slave device.
ESPI_CS0#/ LFRAME#/ GPP_A5	O	eSPI Chip Select 0: Driving CS# signal low to select eSPI slave for the transaction.
ESPI_RESET#/ SUS_STAT#/ GPP_A14	O	eSPI Reset: Reset signal from the PCH to eSPI slave.



17.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
ESPI_CLK	Pull-down	20 kohm +/- 30%	
ESPI_IO[3:0]	Pull-up	20 kohm +/- 30%	
ESPI_CS [0]#	Pull-up	20 kohm +/- 30%	

17.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset (Note 1)	Immediately after Reset (Note 1)	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 [1:0]#	Primary	Internal Pull-up	Driven High	Driven High	Off
ESPI_ALERT [1:0]#	Primary	Internal Pull-up	Driven High	Driven High	Off
ESPI_RESET#	Primary	Driven Low	Driven High	Driven High	Off
Note: 1. Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.					

17.6 Functional Description

17.6.1 Features

The PCH eSPI controller supports the following features:

- Support for Master Attached Flash and Slave Attached Flash.
- Support for 20 MHz, 24 MHz, 30 MHz, 48 MHz, and 60 MHz (configured by soft straps)
- 1.8V support only
- Up to quad mode support
- In-band messages for communication between the PCH and slave device to eliminate side-band signals
- Real time SPI flash sharing, allowing real time operational access by the PCH and slave device Transmitting RTC time/date to the slave device upon request

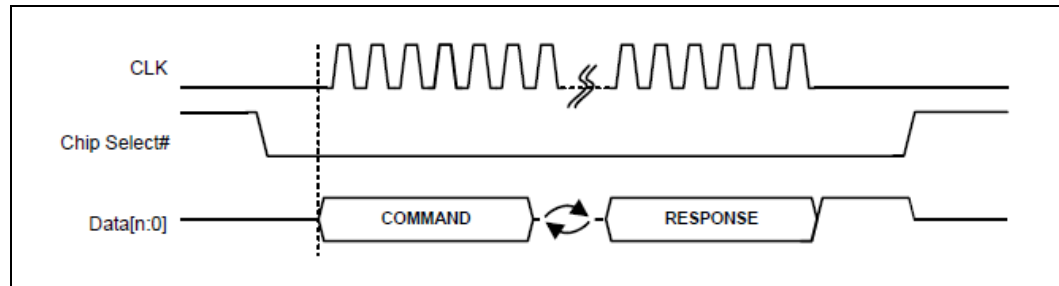
Note: For PCH SKU with two CS#, the PCH eSPI supports two ALERT# pins to support alerts from two supported slave devices to the PCH. However, note that if only a single slave is connected, the eSPI Compatibility Spec requires that the Slave must operate with in-band Alert# signaling in order to free up the GPIO pin required for the discrete Alert# pin.

17.6.2 Protocols

Below is an overview of the basic eSPI protocol. Refer to the latest eSPI Specification and corresponding platform eSPI Compatibility Specification for more details.



Figure 17-1. 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).

17.6.3 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.

17.6.4 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.

17.6.5 Slave Discovery

The PCH eSPI interface is enabled using a hard pin strap. If this strap is asserted (high) at RSMRST# de-assertion, the eSPI controller is enabled and assumes that a slave is connected to the interface. The controller does not perform any other discovery to confirm the presence of the slave connection.

If the ESPI_EN HW strap is de-asserted (low), the eSPI controller will gate all its clocks and put itself to sleep.



17.6.6 Flash Sharing Mode

eSPI supports both Master and Slave Attached Flash sharing (abbreviated in this as MAFS and SAFS, respectively). The Flash sharing mode selected for a specific platform is dependent on strap settings.

In order for SAFS to work, the Slave must support the Flash Access channel.

17.6.7 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.

Table 17-1 summarizes the eSPI channels and supported transactions.

Table 17-1. 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

17.6.7.1 Peripheral Channel (Channel 0) Overview

The Peripheral channel performs the following Functions:

- Target for PCI Device D31:F0: The eSPI controller duplicates the legacy LPC PCI Configuration space registers. These registers are mostly accessed via the BIOS, though some are accessed via the OS as well.
- Tunnel all Host to eSPI slave (EC/SIO) debug device accesses: these are the accesses that used to go over the LPC bus. These include various programmable and fixed I/O ranges as well as programmable Memory ranges. The programmable ranges and their enables reside in the PCI Configuration space.
- Tunnel all accesses from the eSPI slave to the Host. These include Memory Reads and Writes.

17.6.7.2 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.

17.6.7.2.1 eSPI Virtual Wires (VW)

Below table summarizes the PCH virtual wires in eSPI mode. Refer to the eSPI Compatibility Specification for details.

Table 17-2. eSPI Virtual Wires (VW)

Virtual Wire	PCH Pin Direction	Reset Control	Pin Retained in PCH (For Use by Other Components)
SUS_STAT#	Output	ESPI_RESET#	No
SUSWARN#	Output	ESPI_RESET#	No
SUS_ACK	Input	ESPI_RESET#	No
SUSPWRDNACK	Output	ESPI_RESET#	No
PLTRST#	Output	ESPI_RESET#	Yes
PME# (eSPI Peripheral PME)	Input	ESPI_RESET#	N/A
WAKE#	Input	ESPI_RESET#	No
SMI#	Input	PLTRST#	N/A
SCI#	Input	PLTRST#	N/A
RCIN#	Input	PLTRST#	No
SLP_A#	Output	ESPI_RESET#	Yes
SLP_S3#/SLP_S4#/ SLP_S5#/SLP_LAN#/ SLP_WLAN#	Output	DSW_PWROK	Yes
SLAVE_BOOT_LOAD_DONE	Input	ESPI_RESET#	N/A
SLAVE_BOOT_LOAD_STATUS	Input	ESPI_RESET#	N/A
HOST_RST_WARN	Output	PLTRST#	N/A
HOST_RST_ACK	Input	PLTRST#	N/A
OOB_RST_WARN	Output	ESPI_RESET#	N/A
OOB_RST_ACK	Input	ESPI_RESET#	N/A
HOST_C10	Output	PLTRST#	N/A
ERROR_NONFATAL	Input	ESPI_RESET#	N/A
ERROR_FATAL	Input	ESPI_RESET#	N/A

17.6.7.2.2 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.

17.6.7.3 Out-of-Band Channel (Channel 2) Overview

The Out-of-Band channel performs the following Functions:

- Tunnel MCTP Packets between the Intel® CSME and eSPI slave device: The Intel® CSME communicates MCTP messages to/from the device by embedding those packets over the eSPI protocol. This eliminates the SMBus connection between the PCH and the slave device which was used to communicate the MCTP messages in prior PCH generations. The eSPI controller simply acts as a message transport and forwards the packets between the Intel® CSME and eSPI device.
- 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.

17.6.7.3.1 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 [Figure 17-2](#) and [Figure 17-3](#).

Figure 17-2. eSPI Slave Request to PCH for PCH Temperature

eSPI Slave to PCH: Request 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	Dest Slave Addr [7:1] = 01h (PCH OOB HW Handler)						0
4	Command Code = 01h (Get_PCH_Temp)						
5	Byte Count = 01h						
6	Source Slave Address [7:0] = 0Fh (eSPI Slave 0/EC)						1



Figure 17-3. PCH Response to eSPI Slave with PCH Temperature

PCH to eSPI Slave: Response 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	Dest Slave Addr [7:0] = 0Eh (eSPI slave 0/EC)						0	
4	Command Code = 01h (Get_PCH_Temp)							
5	Byte Count = 02h							
6	Source Slave Address [7:1] = 01h (PCH OOB HW Handler)						1	
7	PCH Temperature Data [7:0]							

17.6.7.3.2 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 [Figure 17-4](#) and [Figure 17-5](#).

Figure 17-4. eSPI Slave Request to PCH for PCH RTC Time

eSPI Slave to PCH: Request for RTC Time								
Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]			Length[11:8] = 0h				
2	Length[7:0] = 04h							
3	Dest Slave Addr [7:1] = 01h (PCH OOB HW Handler)						0	
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 01h							
6	Source Slave Address [7:0] = 0Fh (eSPI Slave 0/EC)						1	

Figure 17-5. PCH Response to eSPI Slave with RTC Time

PCH to eSPI Slave: Response with 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] = 0Ch							
3	Dest Slave Addr [7:0] = 0Eh (eSPI slave 0/EC)							0
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 09h							
6	Source Slave Address [7:1] = 01h (PCH OOB HW Handler)							1
7	Reserved			DM	HF	DS		
8	PCH RTC Time: Seconds							
9	PCH RTC Time: Minutes							
10	PCH RTC Time: Hours							
11	PCH RTC Time: Day of Week							
12	PCH RTC Time: Day of Month							
13	PCH RTC Time: Month							
14	PCH RTC Time: Year							

Notes:

- 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.
- 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.
- 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.

17.6.7.4 Flash Access Channel (Channel 3) Overview

Master Attached Flash (MAF): this is the configuration where the flash device is directly attached to the PCH and it is the only configuration supported for client platforms. This configuration allows the eSPI device to access the flash device attached to the PCH through a set of flash access commands. These commands are routed to the flash controller and the return data is sent back to the eSPI device.

The Master Attached Flash Channel controller (MAFCC) tunnels flash accesses from eSPI slave to the PCH flash controller. The MAFCC simply provides Flash Cycle Type, Address, Length, Payload (for writes) to the flash controller. The flash controller is responsible for all the low level flash operations to perform the requested command and provides a return data/status back to the MAFCC, which then tunnels it back to the eSPI slave in a separate completion packet.

The Slave Attached Flash Channel controller (SAFCC) tunnels flash accesses from the PCH to the BMC flash controller. In the case of Slave Attached Flash Sharing (SAFS), the post-security flash request from the SPI Flash Controller (Flash Cycle Type, Address, Length, Payload [for writes]) is routed to the eSPI SAFCC, which tunnels it forward to the BMC over the eSPI link. The completion from the Flash device access, in a separate command, is returned by the BMC to the eSPI SAFCC, which forwards it back to the SPI flash controller.



17.6.7.4.1 Master Attached Flash Channel Controller (MAFCC) Flash Operations and Addressing

The EC is allocated a dedicated region within the eSPI Master-Attached flash device. The EC has default read, write, and erase access to this region.

The EC can also access any other flash region as permitted by the Flash Descriptor settings. As such, the EC uses linear addresses, valid up to the maximum supported flash size, to access the flash.

The MAFCC supports flash read, write, and erase operations only.

17.6.7.4.2 Slave Attached Flash Channel Controller (SAFCC) Flash Operation and Addressing

The PCH is allocated dedicated regions (for each of the supported masters) within the eSPI slave-attached flash devices. The PCH has read, write, and erase access to these regions, as well as any other regions that maybe permitted by the region protections set in the Flash Descriptor.

The Slave will optionally perform additional checking on the PCH provided address. In case of an error due to incorrect address or any other issues it will synthesize an unsuccessful completion back to the eSPI Master.

The SAFCC supports Flash Read, Write and Erase operations. It also supports the RPMC, Read SFDP and Read JEDEC ID commands.

§ §



18 General Purpose Input and Output (GPIO)

18.1 Acronyms

Acronyms	Description
GPI	General Purpose Input
GPO	General Purpose Output
GPP	General Purpose I/O in Primary Well
GPD	General Purpose I/O in Deep Sleep Well

18.2 Overview

The PCH General Purpose Input/Output (GPIO) signals are grouped into multiple groups (GPP_A, GPP_B, GPP_C, GPP_D, GPP_E, GPP_F, GPP_G, GPP_H, GPP_I, GPP_J, GPP_K, and GPD group), and are powered by either the PCH Primary well or Deep Sleep well.

The high level features of GPIO:

- Per-group configurable 3.3V or 1.8V voltage (except for GPP_I group and GPD (3.3V only), and GPP_J group (1.8V only))
- Configurable as an input or output signal.
- Configurable GPIO pad ownership by host, ME, or ISH.
- SCI (GPE) and IOAPIC interrupt capable on all GPIOs
- NMI and SMI capability capable (on selected GPIOs).
- PWM, Serial Blink capable (on selected GPIOs).
- Programmable hardware debouncer (on GPD3/PWRBTN# pin)

Table 18-1. GPIO Group Summary (Sheet 1 of 2)

GPIO Group	Power Pins	Voltage
Primary Well Group A (GPP_A)	VCCPGPPA	1.8V or 3.3V
Primary Well Group B (GPP_B) Primary Well Group C (GPP_C)	VCCPGPPBC	1.8V or 3.3V
Primary Well Group D (GPP_D)	VCCPGPPD	1.8V or 3.3V
Primary Well Group E (GPP_E) Primary Well Group F (GPP_F)	VCCPGPPEF	1.8V or 3.3V
Primary Well Group G (GPP_G)	VCCPGPPG_3P3 or VCCPRIM_1P8	1.8V or 3.3V
Primary Well Group H (GPP_H) Primary Well Group K (GPP_K)	VCCPGPPHK	1.8V or 3.3V

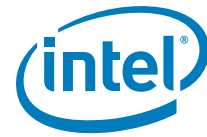


Table 18-1. GPIO Group Summary (Sheet 2 of 2)

GPIO Group	Power Pins	Voltage
Primary Well Group I (GPP_I)	VCCPRIM_3P3	3.3V Only
Primary Well Group J (GPP_J)	VCCPRIM_1P8	1.8V Only
Deep Sleep Well Group (GPD)	VCCDSW_3P3	3.3V Only

Note: Except for GPP_G group, the operating voltage of a GPIO group having voltage configurability (3.3V or 1.8V) is selected by both connecting the corresponding power pin and setting the group-voltage-selection soft strap to the desired voltage. GPP_G supports dynamic voltage configuration ONLY when the family is used for SDXC functionality. If the family is used for GPIOs, per pin voltage configuration is NOT supported. GPP_G group voltage is selected by setting the corresponding soft strap only.

18.3 Signal Description

Table 18-2 summarizes the GPIO implementation in the PCH.

Table 18-2. General Purpose I/O Signals (Sheet 1 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
Primary Well Group A (Per-group 1.8 V or 3.3 V)																
GPP_A0	VCCPGPPA	RCIN#	in			ESPI_ALER T1#	in			Native F1/ Native F3/ GP-In		no	yes(1)	Native F1: Z / Native F3: PU / GPIO: Z	Native F1: Z / Native F3: PU / GPIO: Z	Native F1 if eSPI or LPC pin strap = 0 (LPC mode). Else, GPI in Non-server SKUs and Native F3 in Server SKUs
GPP_A1	VCCPGPPA	LAD0	inout			ESPI_IO0	inout			Native F1/ Native F3		no	yes(1)	Native F1: PU / Native F3: PU	Native F1: PU / Native F3: PU	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A2	VCCPGPPA	LAD1	inout			ESPI_IO1	inout			Native F1/ Native F3		no	yes(1)	Native F1: PU / Native F3: PU	Native F1: PU / Native F3: PU	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A3	VCCPGPPA	LAD2	inout			ESPI_IO2	inout			Native F1/ Native F3		no	yes(1)	Native F1: PU / Native F3: PU	Native F1: PU / Native F3: PU	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A4	VCCPGPPA	LAD3	inout			ESPI_IO3	inout			Native F1/ Native F3		no	yes(1)	Native F1: PU / Native F3: PU	Native F1: PU / Native F3: PU	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A5	VCCPGPPA	LFRAME#	out			ESPI_CS0#	out			Native F1/ Native F3		no	yes(2)	Native F1: H / Native F3: PU	Native F1: H / Native F3: H	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A6	VCCPGPPA	SERIRQ	inout			ESPI_CS1#	out			Native F1/ Native F3/ GP-In		no	yes(1)	Native F1: Z / Native F3: PU / GPIO: Z	Native F1: Z / Native F3: H / GPIO: Z	Native F1 if eSPI or LPC pin strap = 0 (LPC mode). Else, GPI in Non-server SKUs and Native F3 in Server SKUs
GPP_A7	VCCPGPPA	PIRQA#	iod			ESPI_ALER T0#	in			Native F1/ Native F3/ GP-In		no	yes(1)	Native F1: Z / Native F3: PU / GPIO: Z	Native F1: Z / Native F3: PU / GPIO: Z	Native F1 if eSPI or LPC pin strap = 0 (LPC mode). Else, GPI in Non-server SKUs and Native F3 in Server SKUs
GPP_A8	VCCPGPPA	CLKRUN#	iod							Native F1/ GP-In		no	yes(1)	Native F1: Z / GPIO: Z	Native F1: Z / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI in non-server SKUs and native F1 in Server SKUs. For Server SKUs, when eSPI pin strap set to eSPI, GPIO functionality is not available.
GPP_A9	VCCPGPPA	CLKOUT_LPC 0	out			ESPI_CLK	out			Native F1/ Native F3		no	yes(1)	Native F1: T / Native F3: PD	Native F1: T / Native F3: L	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3
GPP_A10	VCCPGPPA	CLKOUT_LPC 1	out							Native F1/ GP-In		no	yes(1)	Native F1: T / GPIO: Z	Native F1: T / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI. When eSPI pin strap set to eSPI, for Server SKUs, GPIO functionality is not available.
GPP_A11	VCCPGPPA	PME#	iod	SD_VDD2_P- WR_EN#	out					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI.
GPP_A12	VCCPGPPA	BM_BUSY#	in	ISH_GP6	inout	SX_EX- IT_HOLD OF F#	in			GP-In		no	yes(1)	Z	Z	
GPP_A13	VCCPGPPA	SUSWARN# / SUSPWDNA CK	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: Z	Native F1: L / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI.



Table 18-2. General Purpose I/O Signals (Sheet 2 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_A14	VCCPGPPA	SUS_STAT#	out			ESPI_RESE_T#	out			Native F1/ Native F3		no	yes(2)	Native F1: L / Native F3: L	Native F1: note / Native F3: H	Native F1 if eSPI or LPC pin strap = LPC. Else, Native F3 SUS_STAT# state After Reset: L, and then H after PCH_PWROK asserts
GPP_A15	VCCPGPPA	SUSACK#	in							Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI.
GPP_A16	VCCPGPPA	CLKOUT_48	out							Native F1/ GP-In		no	yes(1)	Native F1: T / GPIO: Z	Native F1: T / GPIO: Z	Native F1 if eSPI or LPC pin strap = LPC. Else, GPI.
GPP_A17	VCCPGPPA	SD_VDD1_P WR_EN#	out	ISH_GP7	inout					GP-In		no	yes(1)	Z	Z	
GPP_A18	VCCPGPPA	ISH_GP0	inout							GP-In		no	yes(1)	Z	Z	
GPP_A19	VCCPGPPA	ISH_GP1	inout							GP-In		no	yes(1)	Z	Z	
GPP_A20	VCCPGPPA	ISH_GP2	inout							GP-In		no	yes(1)	Z	Z	
GPP_A21	VCCPGPPA	ISH_GP3	inout							GP-In		no	yes(1)	Z	Z	
GPP_A22	VCCPGPPA	ISH_GP4	inout							GP-In		no	yes(1)	Z	Z	
GPP_A23	VCCPGPPA	ISH_GP5	inout							GP-In		no	yes(1)	Z	Z	
Primary Well Group B (Per-group 1.8 V or 3.3 V)																
GPP_B0	VCCPGPPBC	GSPI0_CS1#	out							GP-In		no	yes(1)	Z	Z	
GPP_B1	VCCPGPPBC	GSPI1_CS1#	out	TIME_SYNC1	inout					GP-In		no	yes(1)	Z	Z	
GPP_B2	VCCPGPPBC	VRALERT#	in							GP-In		no	yes(1)	Z	Z	
GPP_B3	VCCPGPPBC	CPU_GP2	in							GP-In		no	yes(1)	Z	Z	
GPP_B4	VCCPGPPBC	CPU_GP3	in							GP-In		no	yes(1)	Z	Z	
GPP_B5	VCCPGPPBC	SRCLKREQ_0#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B6	VCCPGPPBC	SRCLKREQ_1#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B7	VCCPGPPBC	SRCLKREQ_2#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B8	VCCPGPPBC	SRCLKREQ_3#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B9	VCCPGPPBC	SRCLKREQ_4#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B10	VCCPGPPBC	SRCLKREQ_5#	iod							GP-In		no	yes(1)	Z	Z	
GPP_B11	VCCPGPPBC	I2S_MCLK	out							GP-In		no	yes(1)	Z	Z	
GPP_B12	VCCPGPPBC	SLP_S0#	out							Native F1		no	yes(1)	H (Refer note)	H	The During Reset Pin State of this GPIO is H via a ~20 k pull-up to 3.3 V independent of soft strap assigned pad voltage. A 1.8 V device connected to this GPIO must be capable of taking ~20 kohm pull-up to 3.3V.
GPP_B13	VCCPGPPBC	PLTRST#	out							Native F1		no	yes(2)	L	H	
GPP_B14	VCCPGPPBC	SPKR	out							GP-Out	Top Swap Override	no	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.
GPP_B15	VCCPGPPBC	GSPI0_CS0#	out							GP-In		no	yes(1)	Z	Z	
GPP_B16	VCCPGPPBC	GSPI0_CLK	out							GP-In		no	yes(1)	Z	Z	
GPP_B17	VCCPGPPBC	GSPI0_MISO	in							GP-In		no	yes(1)	Z	Z	
GPP_B18	VCCPGPPBC	GSPI0_MOSI	out							GP-Out	No Reboot	no	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.
GPP_B19	VCCPGPPBC	GSPI1_CS0#	out							GP-In		no	yes(1)	Z	Z	
GPP_B20	VCCPGPPBC	GSPI1_CLK	out							GP-In		no	yes(1)	Z	Z	
GPP_B21	VCCPGPPBC	GSPI1_MISO	in							GP-In		no	yes(1)	Z	Z	
GPP_B22	VCCPGPPBC	GSPI1_MOSI	out							GP-Out	Boot BIOS Strap Bit	no	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.

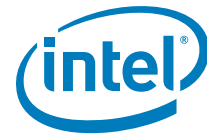


Table 18-2. General Purpose I/O Signals (Sheet 3 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_B23	VCCPGPPBC	SML1ALERT#	iod	PCHHOT#	od					GP-Out	Intel® DCI-OOB Enable	yes	no	PD	L	Strap read at rising edge of RSMRST#. The internal 20 kohm ± 30% Pull-down is disabled after RSMRST# de-asserts. Note: When used as PCHHOT# and strap low, a 150k weak board Pull-up is recommended to ensure it does not override the internal Pull-down strap sampling.
Primary Well Group C (Per-group 1.8 V or 3.3 V)																
GPP_C0	VCCPGPPBC	SMBCLK	iod							Native F1		yes	yes(1)	Z	Z	
GPP_C1	VCCPGPPBC	SMBDATA	iod							Native F1		yes	yes(1)	Z	Z	
GPP_C2	VCCPGPPBC	SMBALERT#	iod							Native F1/ GP-Out	TLS Confidentiality	yes	no	Native F1: PD / GPIO: PD	Native F1: L / GPIO: L	Native F1 if Intel® SMBus ASD Mode Configuration soft strap = 1, else GP-Out; Strap read at rising edge of RSMRST#. The internal 20 kohm ± 30% Pull-down is disabled after RSMRST# de-asserts.
GPP_C3	VCCPGPPBC	SML0CLK	iod							Native F1		yes	yes(1)	Z	Z	
GPP_C4	VCCPGPPBC	SML0DATA	iod							Native F1		yes	yes(1)	Z	Z	
GPP_C5	VCCPGPPBC	SMLOALERT#	iod							GP-Out	eSPI or LPC	yes	no	PD	L	Strap read at rising edge of RSMRST#. The internal 20 kohm ± 30% Pull-down is disabled after RSMRST# de-asserts.
GPP_C6	VCCPGPPBC	SML1CLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C7	VCCPGPPBC	SML1DATA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C8	VCCPGPPBC	UART0A_RXD	in							GP-In		no	yes(1)	Z	Z	
GPP_C9	VCCPGPPBC	UART0A_TXD	out							GP-In		no	yes(1)	Z	Z	
GPP_C10	VCCPGPPBC	UART0A_RTS#	out							GP-In		no	yes(1)	Z	Z	
GPP_C11	VCCPGPPBC	UART0A_CTS#	in							GP-In		no	yes(1)	Z	Z	
GPP_C12	VCCPGPPBC	UART1_RXD	in	ISH_UART1_RXD	in					GP-In		no	yes(1)	Z	Z	
GPP_C13	VCCPGPPBC	UART1_TXD	out	ISH_UART1_TXD	out					GP-In		no	yes(1)	Z	Z	
GPP_C14	VCCPGPPBC	UART1_RTS#	out	ISH_UART1_RTS#	out					GP-In		no	yes(1)	Z	Z	
GPP_C15	VCCPGPPBC	UART1_CTS#	in	ISH_UART1_CTS#	in					GP-In		no	yes(1)	Z	Z	
GPP_C16	VCCPGPPBC	I2C0_SDA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C17	VCCPGPPBC	I2C0_SCL	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C18	VCCPGPPBC	I2C1_SDA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C19	VCCPGPPBC	I2C1_SCL	iod							GP-In		yes	yes(1)	Z	Z	
GPP_C20	VCCPGPPBC	UART2_RXD	in							GP-In		no	yes(1)	Z	Z	
GPP_C21	VCCPGPPBC	UART2_TXD	out							GP-In		no	yes(1)	Z	Z	
GPP_C22	VCCPGPPBC	UART2_RTS#	out							GP-In		no	yes(1)	Z	Z	
GPP_C23	VCCPGPPBC	UART2_CTS#	in							GP-In		no	yes(1)	Z	Z	
Primary Well Group D (Per-group 1.8 V or 3.3 V)																
GPP_D0	VCCPGPPD	SPI1_CS#	out			SBK0	od	BK0	od	GP-In		no	yes(1)	Z	Z	
GPP_D1	VCCPGPPD	SPI1_CLK	out			SBK1	od	BK1	od	GP-In		no	yes(1)	Z	Z	
GPP_D2	VCCPGPPD	SPI1_MISO	inout			SBK2	od	BK2	od	GP-In		no	yes(1)	Z	Z	
GPP_D3	VCCPGPPD	SPI1_MOSI	inout			SBK3	od	BK3	od	GP-In		no	yes(1)	Z	Z	
GPP_D4	VCCPGPPD	ISH_I2C2_SDA	iod	I2C3_SDA	iod	SBK4	od	BK4	od	GP-In		yes	yes(1)	Z	Z	
GPP_D5	VCCPGPPD	I2S2_SFRM	inout			CNV_RF_ESET#	out			Native F3		no	yes(2)	Z	H	
GPP_D6	VCCPGPPD	I2S2_TXD	out			MODEM_CLKREQ	out			Native F3		no	yes(1)	Z	H	
GPP_D7	VCCPGPPD	I2S2_RXD	in							GP-In		no	yes(1)	Z	Z	



Table 18-2. General Purpose I/O Signals (Sheet 4 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_D8	VCCPGPPD	I2S2_SCLK	inout							GP-In		no	yes(1)	Z	Z	
GPP_D9	VCCPGPPD	ISH_SPI_CS#	out			GSPI2_CS0#	out			GP-In		no	yes(1)	Z	Z	
GPP_D10	VCCPGPPD	ISH_SPI_CLK	out			GSPI2_CLK	out			GP-In		no	yes(1)	Z	Z	
GPP_D11	VCCPGPPD	ISH_SPI_MISO	in	GP_BSSB_CLK	in	GSPI2_MISO	in			Native F2		no	yes(1)	Z	Z	
GPP_D12	VCCPGPPD	ISH_SPI_MOSI	out	GP_BSSB_DI	in	GSPI2_MOSI	out			Native F2		no	yes(1)	Z	Z	
GPP_D13	VCCPGPPD	ISH_UART0_RXD	in			I2C2_SDA	iod			GP-In		no	yes(1)	Z	Z	
GPP_D14	VCCPGPPD	ISH_UART0_TXD	out			I2C2_SCL	iod			GP-In		no	yes(1)	Z	Z	
GPP_D15	VCCPGPPD	ISH_UART0_RTS#	out	GSPI2_CS1#	out			CNV_WFEN	in	GP-In		no	yes(1)	Z	Z	
GPP_D16	VCCPGPPD	ISH_UART0_CTS#	in					CNV_WCEN	in	GP-In		no	yes(1)	Z	Z	
GPP_D17	VCCPGPPD	DMIC_CLK1	out	SNDW3_CLK	inout					GP-In		no	yes(1)	Z	Z	
GPP_D18	VCCPGPPD	DMIC_DATA1	in	SNDW3_DATA	inout					GP-In		no	yes(1)	Z	Z	
GPP_D19	VCCPGPPD	DMIC_CLK0	out	SNDW4_CLK	inout					GP-In		no	yes(1)	Z	Z	
GPP_D20	VCCPGPPD	DMIC_DATA0	in	SNDW4_DATA	inout					GP-In		no	yes(1)	Z	Z	
GPP_D21	VCCPGPPD	SPI1_IO2	inout							GP-In		no	yes(1)	Z	Z	
GPP_D22	VCCPGPPD	SPI1_IO3	inout							GP-In		no	yes(1)	Z	Z	
GPP_D23	VCCPGPPD	ISH_I2C2_SCL	iod	I2C3_SCL	iod					GP-In		yes	yes(1)	Z	Z	
Primary Well Group E (Per-group 1.8 V or 3.3 V)																
GPP_E0	VCCPGPPEF	SATAXPCEI0	in	SATAGP0	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 0 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_E1	VCCPGPPEF	SATAXPCEI1	in	SATAGP1	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 1 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_E2	VCCPGPPEF	SATAXPCEI2	in	SATAGP2	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 2 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_E3	VCCPGPPEF	CPU_GP0	in				out			GP-In		no	yes(1)	Z	Z	
GPP_E4	VCCPGPPEF	SATA_DEVSLP0	od				out			GP-In		no	yes(1)	Z	Z	
GPP_E5	VCCPGPPEF	SATA_DEVSLP1	od				out			GP-In		no	yes(1)	Z	Z	
GPP_E6	VCCPGPPEF	SATA_DEVSLP2	od				out			GP-In		no	yes(1)	Z	Z	
GPP_E7	VCCPGPPEF	CPU_GP1	in				out			GP-In		no	yes(1)	Z	Z	
GPP_E8	VCCPGPPEF	SATALED#	od				out			GP-In		no	yes(1)	Z	Z	
GPP_E9	VCCPGPPEF	USB2_OC0#	in				out			GP-In		no	yes(1)	Z	Z	
GPP_E10	VCCPGPPEF	USB2_OC1#	in				out			GP-In		no	yes(1)	Z	Z	
GPP_E11	VCCPGPPEF	USB2_OC2#	in				out			GP-In		no	yes(1)	Z	Z	
GPP_E12	VCCPGPPEF	USB2_OC3#	in				out			GP-In		no	yes(1)	Z	Z	

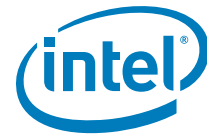


Table 18-2. General Purpose I/O Signals (Sheet 5 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
Primary Well Group F (Per-group 1.8 V or 3.3 V)																
GPP_F0	VCCPGPPEF	SATAXPCE3	in	SATAGP3	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 3 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_F1	VCCPGPPEF	SATAXPCE4	in	SATAGP4	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 4 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_F2	VCCPGPPEF	SATAXPCE5	in	SATAGP5	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 5 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_F3	VCCPGPPEF	SATAXPCE6	in	SATAGP6	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 6 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_F4	VCCPGPPEF	SATAXPCE7	in	SATAGP7	in					Native F1/ GP-In		no	yes(1)	Native F1: PU / GPIO: Z	Native F1: PU / GPIO: Z	Default Native F1 is set by SATA / PCIe GP Select for Port 7 soft strap. Note: Different Connectors have different polarity for the SATAxPCIe select. Note: Default state is GP-In before soft straps are loaded or in the case the soft straps are invalid
GPP_F5	VCCPGPPEF	SATA_DEVSL P3	od							GP-In		no	yes(1)	Z	Z	
GPP_F6	VCCPGPPEF	SATA_DEVSL P4	od							GP-In		no	yes(1)	Z	Z	
GPP_F7	VCCPGPPEF	SATA_DEVSL P5	od							GP-In		no	yes(1)	Z	Z	
GPP_F8	VCCPGPPEF	SATA_DEVSL P6	od							GP-In		no	yes(1)	Z	Z	
GPP_F9	VCCPGPPEF	SATA_DEVSL P7	od							GP-In		no	yes(1)	Z	Z	
GPP_F10	VCCPGPPEF	SATA_SCLOCK	od							GP-In		no	yes(1)	Z	Z	
GPP_F11	VCCPGPPEF	SATA_SLOA D	od							GP-In		no	yes(1)	Z	Z	
GPP_F12	VCCPGPPEF	SATA_S_DATAOUT1	od							GP-In		no	yes(1)	Z	Z	
GPP_F13	VCCPGPPEF	SATA_S_DATAOUT0	od							GP-In		no	yes(1)	Z	Z	
GPP_F14	VCCPGPPEF			PS_ON#	out					Native F2		no	yes(2)	L	L	
GPP_F15	VCCPGPPEF	USB2_OC4#	in							GP-In		no	yes(1)	Z	Z	
GPP_F16	VCCPGPPEF	USB2_OC5#	in							GP-In		no	yes(1)	Z	Z	
GPP_F17	VCCPGPPEF	USB2_OC6#	in							GP-In		no	yes(1)	Z	Z	
GPP_F18	VCCPGPPEF	USB2_OC7#	in							GP-In		no	yes(1)	Z	Z	
GPP_F19	VCCPGPPEF	eDP_VDDEN	out							GP-In		no	yes(1)	Z	Z	
GPP_F20	VCCPGPPEF	eDP_BKLTEN	out							GP-In		no	yes(1)	Z	Z	
GPP_F21	VCCPGPPEF	eDP_BKLTCT L	out							GP-In		no	yes(1)	Z	Z	
GPP_F22	VCCPGPPEF	DDPF_C-TRCLK	iod							GP-In		no	yes(1)	Z	Z	



Table 18-2. General Purpose I/O Signals (Sheet 6 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_F23	VCCPGPPEF	DDPF_C-TRLDATA	iod							GP-Out	Display Port F Detected	no	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm \pm 30% Pull-down is disabled after PCH_PWROK is high.
Primary Well Group G (Per-group 1.8 V or 3.3 V)																
GPP_G0	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_CMD	inout							GP-In		no	yes(1)	Z	Z	
GPP_G1	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_DATA0	inout							GP-In		no	yes(1)	Z	Z	
GPP_G2	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_DATA1	inout							GP-In		no	yes(1)	Z	Z	
GPP_G3	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_DATA2	inout							GP-In		no	yes(1)	Z	Z	
GPP_G4	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_DATA3	inout							GP-In		no	yes(1)	Z	Z	
GPP_G5	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_CD#	in							GP-In		no	yes(1)	Z	Z	
GPP_G6	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_CLK	out							GP-In		no	yes(1)	Z	Z	
GPP_G7	VCCPGPPG_3P3 or VCCPRIM_1 P8	SD_WP	in							GP-In		no	yes(1)	Z	Z	
Primary Well Group H (Per-group 1.8 V or 3.3 V)																
GPP_H0	VCCPGPPHK	SRCLKREQ_6#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H1	VCCPGPPHK	SRCLKREQ_7#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H2	VCCPGPPHK	SRCLKREQ_8#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H3	VCCPGPPHK	SRCLKREQ_9#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H4	VCCPGPPHK	SRCLKREQ_10#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H5	VCCPGPPHK	SRCLKREQ_11#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H6	VCCPGPPHK	SRCLKREQ_12#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H7	VCCPGPPHK	SRCLKREQ_13#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H8	VCCPGPPHK	SRCLKREQ_14#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H9	VCCPGPPHK	SRCLKREQ_15#	iod							GP-In		no	yes(1)	Z	Z	
GPP_H10	VCCPGPPHK	SML2CLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H11	VCCPGPPHK	SML2DATA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H12	VCCPGPPHK	SML2ALERT_#	iod							GP-Out	eSPI Flash Sharing Mode	yes	no	PD	L	Strap read at rising edge of RSMRST#. The internal 20 kohm \pm 30% Pull-down is disabled after RSMRST# de-asserts.
GPP_H13	VCCPGPPHK	SML3CLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H14	VCCPGPPHK	SML3DATA	iod							GP-In		yes	yes(1)	Z	Z	



Table 18-2. General Purpose I/O Signals (Sheet 7 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_H15	VCCPGPPHK	SML3ALERT#	iod							GP-In	Reserved	yes	yes(1)	Z	Z	Strap read at rising edge of RSMRST#. External pull-up is required. Recommend 100 kohm if pulled up to 3.3 V or 75 kohm 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.
GPP_H16	VCCPGPPHK	SML4CLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H17	VCCPGPPHK	SML4DATA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H18	VCCPGPPHK	SML4ALERT#	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H19	VCCPGPPHK	ISH_I2C0_SDA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H20	VCCPGPPHK	ISH_I2C0_SCL	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H21	VCCPGPPHK	ISH_I2C1_SDA	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H22	VCCPGPPHK	ISH_I2C1_SCL	iod							GP-In		yes	yes(1)	Z	Z	
GPP_H23	VCCPGPPHK	TIME_SYNC0	inout							GP-In		no	yes(1)	Z	Z	
Primary Well Group I (3.3 V Only)																
GPP_I0	VCCPRIM_3P3	DDPB_HPD0	in	DISP_MISC0	inout					GP-In		no	yes(1)	Z	Z	
GPP_I1	VCCPRIM_3P3	DDPC_HPD1	in	DISP_MISC1	inout					GP-In		no	yes(1)	Z	Z	
GPP_I2	VCCPRIM_3P3	DDPD_HPD2	in	DISP_MISC2	inout					GP-In		no	yes(1)	Z	Z	
GPP_I3	VCCPRIM_3P3	DDPF_HPD3	in	DISP_MISC3	inout					GP-In		no	yes(1)	Z	Z	
GPP_I4	VCCPRIM_3P3	EDP_HPD	in	DISP_MISC4	inout					GP-In		no	yes(1)	Z	Z	
GPP_I5	VCCPRIM_3P3	DDPB_CTRCLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_I6	VCCPRIM_3P3	DDPB_CTRLDATA	iod							GP-Out	Display Port B Detected	yes	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.
GPP_I7	VCCPRIM_3P3	DDPC_CTRCLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_I8	VCCPRIM_3P3	DDPC_CTRLDATA	iod							GP-Out	Display Port C Detected	yes	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.
GPP_I9	VCCPRIM_3P3	DDPD_CTRCLK	iod							GP-In		yes	yes(1)	Z	Z	
GPP_I10	VCCPRIM_3P3	DDPD_CTRLDATA	iod							GP-Out	Display Port D Detected	yes	no	PD	L	Strap read at rising edge of PCH_PWROK. The internal 20 kohm ± 30% Pull-down is disabled after PCH_PWROK is high.
GPP_I11	VCCPRIM_3P3	M2_SKT2_CFG0	in							GP-In		no	yes(1)	Z	Z	
GPP_I12	VCCPRIM_3P3	M2_SKT2_CFG1	in							GP-In		no	yes(1)	Z	Z	
GPP_I13	VCCPRIM_3P3	M2_SKT2_CFG2	in							GP-In		no	yes(1)	Z	Z	
GPP_I14	VCCPRIM_3P3	M2_SKT2_CFG3	in							GP-In		no	yes(1)	Z	Z	
Primary Well Group J (1.8 V Only)																
GPP_J0	VCCPRIM_1P8	CNV_PA_BLA_NKING	in							GP-In		no	yes(1)	Z	Z	
GPP_J1	VCCPRIM_1P8			CPU_C10_GA_TE#	out					Native F2		no	yes(1)	H (Refer note)	H	The During Reset Pin State of this GPIO is H via a ~20 kohm pull-up to 3.3 V independent of soft strap assigned pad voltage. A 1.8 V device connected to this GPIO must be capable of taking ~20 kohm pull-up to 3.3V.



Table 18-2. General Purpose I/O Signals (Sheet 8 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_J2	VCCPRIM_1 P8									GP-In		no	yes(1)	Z	Z	
GPP_J3	VCCPRIM_1 P8									GP-In		no	yes(1)	Z	Z	
GPP_J4	VCCPRIM_1 P8	CNV_BRI_DT	out	UART0B_RTS #	out					Native F1	XTAL Frequency Select	no	no	PD	H	Strap read at rising edge of RSMRST#. The internal 20 kohm ± 30% Pull-down is disabled after RSMRST# de-asserts.
GPP_J5	VCCPRIM_1 P8	CNV_BRI_RS P	in	UART0B_RXD	in					Native F1		no	yes(1)	Z	H	
GPP_J6	VCCPRIM_1 P8	CNV_RGI_D T	out	UART0B_TXD	out					Native F1	M.2 CNV Mode	no	no	Z	H	Strap read at rising edge of RSMRST#. An external pull-up or pull-down resistor is required.
GPP_J7	VCCPRIM_1 P8	CNV_RGI_R SP	in	UART0B_CTS #	in					Native F1		no	yes(1)	Z	H	
GPP_J8	VCCPRIM_1 P8	CNV_M-FUART2_RX D	in							GP-In		no	yes(1)	Z	Z	
GPP_J9	VCCPRIM_1 P8	CNV_M-FUART2_TXD	out							GP-Out	1.8 V VCCSPI	no	no	PD	L	Strap read at rising edge of RSMRST#. The internal 20 kohm ± 30% Pull-down is disabled after RSMRST# de-asserts.
GPP_J10	VCCPRIM_1 P8									GP-In		no	yes(1)	Z	Z	
GPP_J11	VCCPRIM_1 P8	A4WP_PRES ENT	in							Native F1		no	yes(1)	Z	Z	
Primary Well Group K (Per-group 1.8 V or 3.3 V)																
GPP_K0	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K1	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K2	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K3	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K4	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K5	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K6	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K7	VCCPGPPHK									GP-In		yes	yes(1)	Z	Z	
GPP_K8	VCCPGPPHK	Reserved	od							Native F1		no	yes(1)	Z	Z	
GPP_K9	VCCPGPPHK	Reserved	od							Native F1		no	yes(1)	Z	Z	
GPP_K10	VCCPGPPHK	Reserved	od							Native F1		no	yes(1)	Z	Z	
GPP_K11	VCCPGPPHK	Reserved	od							Native F1		no	yes(1)	Z	Z	
GPP_K12	VCCPGPPHK	GSXDOUT	out							GP-In		no	yes(1)	Z	Z	
GPP_K13	VCCPGPPHK	GSXSLOAD	out							GP-In		no	yes(1)	Z	Z	
GPP_K14	VCCPGPPHK	GSXDIN	in							GP-In		no	yes(1)	Z	Z	
GPP_K15	VCCPGPPHK	GSXSRESET #	out							GP-In		no	yes(1)	Z	Z	
GPP_K16	VCCPGPPHK	GSXCLK	out							GP-In		no	yes(1)	Z	Z	
GPP_K17	VCCPGPPHK	ADR_COMPLETE	out							GP-In		no	yes(1)	Z	Z	
GPP_K18	VCCPGPPHK	NMI#	od							GP-In		no	yes(1)	Z	Z	
GPP_K19	VCCPGPPHK	SMI#	od							GP-In		no	yes(1)	Z	Z	
GPP_K20	VCCPGPPHK	Reserved	out							Native F1		no	yes(1)	H (Refer note)	H	The During Reset Pin State of this GPIO is H via a ~20 kohm pull-up to 3.3 V independent of soft strap assigned pad voltage. A 1.8 V device connected to this GPIO must be capable of taking ~20 kohm pull-up to 3.3V.
GPP_K21	VCCPGPPHK	Reserved	out							Native F1		no	yes(1)	H (Refer note)	H	The During Reset Pin State of this GPIO is H via a ~20 kohm pull-up to 3.3 V independent of soft strap assigned pad voltage. A 1.8 V device connected to this GPIO must be capable of taking ~20 kohm pull-up to 3.3V.



Table 18-2. General Purpose I/O Signals (Sheet 9 of 9)

GPIO	Power Rail	Native Function 1	Native Dir 1	Native Function 2	Native Dir 2	Native Function 3	Native Dir 3	Native Function 4	Native Dir 4	Default	Strap?	Input Deglitch	Output Power Sequence Deglitch	Pin State During Reset	Pin State Immediately After Reset	Notes
GPP_K22	VCCPGPPHK	IMGCLKOUT_0	out							GP-In		no	yes(1)	Z	Z	
GPP_K23	VCCPGPPHK	IMGCLKOUT_1	out							GP-In		no	yes(1)	Z	Z	
Deep Sleep Well Group (3.3 V Only)																
GPD0	VCCDSW_3 P3	BATLOW#	in							Native F1		no	yes(1)	Z	Z	
GPD1	VCCDSW_3 P3	ACPRESENT	in							Native F1		no	yes(1)	Refer note	Z	Pin state during reset: In Deep Sx enabled configurations, pin is PD when PD is enabled (Refer ACPRES_PD_DSX_DIS bit); else Z
GPD2	VCCDSW_3 P3	LAN_WAKE#	in							Native F1		no	yes(1)	Z	Z	
GPD3	VCCDSW_3 P3	PWRBTN#	in							Native F1		yes	yes(1)	PU	PU	
GPD4	VCCDSW_3 P3	SLP_S3#	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: L / GPIO: L	Native if SLP_S3# / GPD4 Signal Configuration soft strap = 0, else GPI
GPD5	VCCDSW_3 P3	SLP_S4#	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: H / GPIO: L	Native if SLP_S4# / GPD5 Signal Configuration soft strap = 0, else GPI
GPD6	VCCDSW_3 P3	SLP_A#	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: H / GPIO: L	Native if SLP_A# / GPD6 Signal Configuration soft strap = 0, else GPI
GPD7	VCCDSW_3 P3									GP-Out	Reserved	no	no	Z	L	Strap read at rising edge of DSW_PWROK. External pull-up is required. Recommend 100 kohm. This strap should sample HIGH. There should NOT be any on-board device driving it to opposite direction during strap sampling.
GPD8	VCCDSW_3 P3	SUSCLK	out							Native F1		no	yes(2)	L	T	
GPD9	VCCDSW_3 P3	SLP_WLAN#	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: L / GPIO: L	Native if SLP_WLAN# / GPD9 Signal Configuration soft strap = 0, else GPI
GPD10	VCCDSW_3 P3	SLP_S5#	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: H / GPIO: L	Native if SLP_S5# / GPD10 Signal Configuration soft strap = 0, else GPI
GPD11	VCCDSW_3 P3	LANPHYC	out							Native F1/ GP-In		no	yes(2)	Native F1: L / GPIO: L	Native F1: L / GPIO: L	Native if LAN PHY Power Control GPD11 Signal Configuration soft strap = 0, else GPI
Deglitch Legend										Pin State Legend						
yes(1) - o/p Hi-Z, no internal weak pull during respective pin power sequencing										H - Driven High						
yes(2) - o/p Hi-Z, with integrated 20 kohm ± 30% pull-down during respective pin power sequencing										L - Driven Low						
										PU - Integrated 20 kohm ± 30% pull-up pulls pin high						
										PD - Integrated 20 kohm ± 30% pull-down pulls pin low						
										Z - Hi-Z						
										T - Toggling						

18.4 Functional Description

18.4.1 GPIO Voltage

The following GPIO groups have the per-group voltage configurability. The voltage of all signals in the group is selected by both connecting the corresponding power supply pin and setting the associated soft strap to the desired voltage (3.3V or 1.8V), except for GPP_G group, whose voltage is selected via soft strap only.

- GPP_A, GPP_B, GPP_C, GPP_D, GPP_E, GPP_F, GPP_G, GPP_H, and GPP_K.

The following GPIO groups have fixed operating voltage. The corresponding power pin to these GPIO groups must be connected to the correct voltage:

- GPP_I and GPD groups: 3.3V only.
- GPP_J group: 1.8V only.



18.4.2 Programmable Hardware Debouncer

Hardware debounce capability is supported on GPD3/PWRBTN# 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 1s.

18.4.3 Integrated Pull-ups and Pull-downs

All GPIOs have programmable internal pull-up/pull-down resistors (20 Kohm) which are disabled by default. The internal pull-up/pull-down for each GPIO can be enabled by BIOS programming the corresponding PAD_CFG_DW1 register. Refer to Volume 2 (Register Info) for more details. The internal pull-up / pull down can only be implemented if the toggle rate of the GPIO is no more than 300 kHz.

Note that certain GPIOs used as pin straps have internal PU/PD enabled during reset by default. Refer [Table 18-2](#) for info on which GPIO has integrated PU/PD enabled during reset.

18.4.4 GPPJ_RCOMP_1P8 Signal

The PCH implements the GPPJ_RCOMP_1P8 as an external bias resistor to ground. A 200 Ohm (+/- 1%) resistor to ground is required on the signal and this single resistor can be shared with SD_1P8_RCOMP and SD_3P3_RCOMP on the platform.

18.4.5 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.

18.4.6 SMI#/SCI and NMI

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

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

- GPP_B14, GPP_B20, GPP_B23
- GPP_C[23:22]
- GPP_D[4:0]
- GPP_E[8:0]
- GPP_I[3:0]
- GPP_G[7:0] (support SMI# only).

18.4.7 Timed GPIO

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.

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.

18.4.8 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 multiplexed 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.

18.4.9 GPIO Ownership

Any PCH GPIO can be owned either by the host or the Intel® ME. The designer can select GPIOs that are required by an Intel® ME feature using the Intel® ME FIT tool (available with Intel® ME FW releases). When selected and controlled by the Intel® ME, those GPIOs cannot be used by the host anymore.





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

19.1 Acronyms

Acronyms	Description
GSPI	Generic Serial Peripheral Interface
LTR	Latency Tolerance Reporting

19.2 Overview

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).

19.3 Signal Description

Name	Type	Description
GSPI0_CS0# /GPP_B15	O	Generic SPI 0 Chip Select
GSPI0_CS1# /GPP_B0	O	Generic SPI 0 Chip Select
GSPI0_CLK /GPP_B16	O	Generic SPI 0 Clock
GSPI0_MISO /GPP_B17	I	Generic SPI 0 MISO
GSPI0_MOSI /GPP_B18	O	Generic SPI 0 MOSI Note: This signal is also utilized as a strap. Refer the pin strap section for more info.
GSPI1_CS0# /GPP_B19	O	Generic SPI 1 Chip Select 0
GSPI1_CS1# / GPP_B1/TIME_SYNC1	O	Generic SPI 1 Chip Select 1
GSPI1_CLK /GPP_B20	O	Generic SPI 1 Clock
GSPI1_MISO /GPP_B21	I	Generic SPI 1 MISO
GSPI1_MOSI /GPP_B22	O	Generic SPI 1 MOSI Note: This signal is also utilized as a strap. Refer the pin strap section for more info.
GSPI2_CS0# /GPP_D9 / ISH_SPI_CS#	O	Generic SPI 2 Chip Select 0
GSPI2_CS1# /GPP_D15/ ISH_UART0_RTS#/CNV_WFEN	O	Generic SPI 2 Chip Select 1
GSPI2_CLK /GPP_D10 / ISH_SPI_CLK	O	Generic SPI 2 Clock
GSPI2_MISO /GPP_D11 / ISH_SPI_MISO/GP_BSSB_CLK	I	Generic SPI 2 MISO
GSPI2_MOSI /GPP_D12 / ISH_SPI_MOSI/GP_BSSB_DI	O	Generic SPI 2 MOSI



19.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
GSPI0_MOSI	Pull Down	20 kohm \pm 30%	Internal Pull Down is not enabled by default.
GSPI1_MOSI	Pull Down	20 kohm \pm 30%	Internal Pull Down is not enabled by default.
GSPI2_MOSI	Pull Down	20 kohm \pm 30%	Internal Pull Down is not enabled by default.
GSPI0_MISO	Pull Down	20 kohm \pm 30%	
GSPI1_MISO	Pull Down	20 kohm \pm 30%	
GSPI2_MISO	Pull Down	20 kohm \pm 30%	

19.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ¹	Immediately after Reset ¹	S3/S4/S5	Deep Sx
GSPI2_CS0#, GSPI2_CS1#, GSPI1_CS0#, GSPI1_CS1#, GSPI0_CS0#, GSPI0_CS1#	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
Notes: 1. Reset reference for primary well pins is RSMRST#.					

19.6 Functional Description

19.6.1 Features

The GSPI interfaces support the following features:

- Full duplex synchronous serial interface
- Support Motorola's SPI protocol
- Operates in master mode only
- Supports bit rates up to 20 Mbits/s
- Supports data size from 4 to 32 bits in length and FIFO depths of 64 entries
- Supports DMA with 128-byte FIFO per channel (up to 64-byte burst)

Note: Slave mode is not supported.



19.6.2 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 value, and must also transfer one FIFO entry per access.

For writes, the PCH 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.

19.6.3 DMA Controller

The GSPI controllers have an integrated DMA controller.

19.6.3.1 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.



19.6.3.2 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.

19.6.4 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 the corresponding reset register to bring the controller from reset state into operational mode.

19.6.5 Power Management

19.6.5.1 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.

19.6.5.2 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.

19.6.6 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.

19.6.7 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.

§ §



20 Intel® Serial I/O Inter-Integrated Circuit (I²C) Controllers

20.1 Acronyms

Acronyms	Description
I ² C	Inter-Integrated Circuit
PIO	Programmed Input/Output
SCL	Serial Clock Line
SDA	Serial Data Line

20.2 References

Specification	Location
The I ² C Bus Specification, Version 6	www.nxp.com/documents/user_manual/UM10204.pdf

20.3 Overview

The PCH implements four I²C controllers for four independent I²C interfaces, I2C0-I2C3. Each interface is a two-wire serial interface consisting of a serial data line (SDA) and a serial clock (SCL).

20.4 Signal Description

Name	Type	Description
I2C0_SDA / GPP_C16	I/OD	I²C Link 0 Serial Data Line External Pull-up resistor is required.
I2C0_SCL / GPP_C17	I/OD	I²C Link 0 Serial Clock Line External Pull-up resistor is required.
I2C1_SDA / GPP_C18	I/OD	I²C Link 1 Serial Data Line External Pull-up resistor is required.
I2C1_SCL / GPP_C19	I/OD	I²C Link 1 Serial Clock Line External Pull-up resistor is required.
I2C2_SDA / GPP_D13/ISH_UART0_RXD	I/OD	I²C Link 2 Serial Data Line External Pull-up resistor is required.
I2C2_SCL / GPP_D14/ISH_UART0_TXD	I/OD	I²C Link 2 Serial Clock Line External Pull-up resistor is required.
I2C3_SDA / ISH_I2C2_SDA / GPP_D4 / SBK4 / BK4	I/OD	I²C Link 3 Serial Data Line External Pull-up resistor is required.
I2C3_SCL / ISH_I2C2_SCL / GPP_D23	I/OD	I²C Link 3 Serial Clock Line External Pull-up resistor is required.



20.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ¹	Immediately after Reset ¹	S3/S4/S5	Deep Sx
I2C[3:0]_SDA	Primary	Undriven	Undriven	Undriven	Off
I2C[3:0]_SCL	Primary	Undriven	Undriven	Undriven	Off
Notes: 1. Reset reference for primary well pins is RSMRST#.					

20.6 Functional Description

20.6.1 Features

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).
- 1.8V or 3.3V support (depending on the voltage supplied to the I²C signal group)
- Master I²C 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 I²C used to share the I²C 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. I²C slave mode is not supported.
2. I²C multi masters is not supported.
3. Simultaneous configuration of Fast Mode and Fast Mode Plus/High speed mode is not supported.
4. I²C General Call is not supported.

20.6.2 Protocols Overview

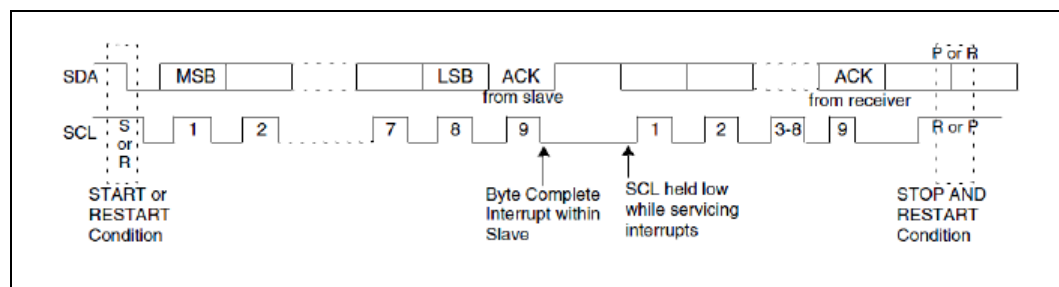
For more information on the I²C protocols and command formats, refer to the industry I²C specification. Below is a simplified description of I²C bus operation:

- 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 I²C 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 20-1. Data Transfer on the I²C Bus



20.6.2.1 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_RESTART_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.

20.6.3 DMA Controller

The I²C controllers 0 to 3 (I²C0 - I²C3) each has an integrated DMA controller.

20.6.3.1 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.

20.6.3.2 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.

20.6.4 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.

20.6.5 Power Management

20.6.5.1 Device Power Down Support

To power down peripherals connected to PCH I²C 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.



20.6.5.2 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.

20.6.6 Interrupts

I²C 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.

20.6.7 Error Handling

Errors that might occur on the external I²C signals are comprehended by the I²C host controller and reported to the I²C bus driver through the MMIO registers.

20.6.8 Programmable SDA Hold Time

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

§ §



21 Gigabit Ethernet Controller

21.1 Acronyms

Acronyms	Description
GbE	Gigabit Ethernet

21.2 References

Specification	Location
Alert Standard Format Specification, Version 1.03	http://www.dmtf.org/standards/asf
IEEE 802.3 Fast Ethernet	https://standards.ieee.org/standard/802_3-2018.html
Intel® Ethernet Connection I219 Datasheet	http://www.intel.com/content/www/us/en/embedded/products/networking/ethernet-connection-i219-datasheet.html

21.3 Overview

The Gigabit Ethernet controller(D31:F6) in conjunction with the Intel® Ethernet Connection I219 provides a complete LAN solution. This chapter describes the behavior of the Gigabit Ethernet Controller. The Gigabit Ethernet Controller can operate at multiple speeds (10/100/1000 Mbps) and in either full duplex or half duplex mode.

21.4 Signal Description

Table 21-1. GbE LAN Signals (Sheet 1 of 2)

Name	Type	Description
PCIE5_TXP PCIE5_TXN PCIE9_TXP PCIE9_TXN PCIE12_TXP PCIE12_TXN PCIE13_TXP PCIE13_TXN	O	Refer to Chapter 26, "PCI Express* (PCIe*)" for details on the PCI Express transmit signals. Note: For PCH, the Intel® Ethernet Connection I219 can be connected to one of the following PCI Express ports 5, 9, 12, or 13
PCIE5_RXP PCIE5_RXN PCIE9_RXP PCIE9_RXN PCIE12_RXP PCIE12_RXN PCIE13_RXP PCIE13_RXN	I	Refer to Chapter 26, "PCI Express* (PCIe*)" for details on the PCI Express transmit signals. Note: For PCH, the Intel® Ethernet Connection I219 can be connected to one of the following PCI Express ports 5, 9, 12, or 13
SML0DATA/GPP_C4	I/OD	Refer to Chapter 30, "System Management Interface and SMLink" for details on the SML0DATA signal. Note: The Intel® Ethernet Connection I219 connects to SML0DATA signal.



Table 21-1. GbE LAN Signals (Sheet 2 of 2)

Name	Type	Description
SML0CLK/GPP_C3	I/OD	Refer to Chapter 30, "System Management Interface and SMLink" for details on the SML0CLK signal. Note: The Intel® Ethernet Connection I219 connects to SML0CLK signal.
LANPHYPC/GPD11	O	LAN PHY Power Control: LANPHYPC should be connected to LAN_DISABLE_N on the PHY. PCH will drive LANPHYPC low to put the PHY into a low power state when functionality is not needed. Note: LANPHYPC can only be driven low if SLP_LAN# is de-asserted. Note: Signal can instead be used as GPD11.
SLP_LAN#	O	LAN Sub-System Sleep Control: If the Gigabit Ethernet Controller is enabled, when SLP_LAN# is de-asserted it indicates that the PHY device must be powered. When SLP_LAN# is asserted, power can be shut off to the PHY device. Note: If Gigabit Ethernet Controller is statically disabled via BIOS, SLP_LAN# will be driven low.
LAN_WAKE#/GPD2	I	LAN WAKE: LAN Wake Indicator from the GbE PHY. Note: LAN_WAKE# functionality is only supported with Intel PHY I219. Connection of a third party LAN device's wake signal to LAN_WAKE# is not supported.

21.5 Integrated Pull-Ups and Pull-Downs

Table 21-2. Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value Ω	Notes
LAN_WAKE#/GPD2	External Pull-up required. Internal Pull-down may be enabled in DeepSx	4.7 k Ω 5%	10 k Ω 5% pull-up resistor is also acceptable

21.6 I/O Signal Planes and States

Table 21-3. Power Plane and States for Output Signals

Signal Name	Power Plane	During Reset ³	Immediately after Reset ³	S3/S4/S5	Deep Sx
LANPHYPC / GPD11	DSW	Undriven	Undriven	Undriven ¹	Undriven ¹
SLP_LAN#	DSW	0/1 ²	0/1 ²	0/1 ²	0/1 ²
Note:					
1. Based on wake events and Intel® CSME state					
2. Configurable based on BIOS settings: '0' When LAN controller is configured as "Disabled" in BIOS, SLP_LAN# will drive "Low"; '1' When LAN controller is configured as "Enabled" in BIOS, SLP_LAN# will drive "High"					
3. Reset reference for DSW well pins is DSW_PWROK.					

Table 21-4. Power Plane and States for Input Signals

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
LAN_WAKE#/GPD2	DSW	Undriven	Undriven	Undriven	Undriven



21.7 Functional Description

The PCH integrates a Gigabit Ethernet (GbE) controller. The integrated GbE controller is compatible with the Intel® Ethernet Connection I219. The integrated GbE controller provides two interfaces for 10/100/1000 Mbps and manageability operation:

- Data link based on PCIe* – A high-speed interface that uses PCIe* electrical signaling at half speed and custom logical protocol for active state operation mode.
- System Management Link (SMLink0)—A low speed connection for low power state mode for manageability communication only. The frequency of this connection can be configured to one of three different speeds (100 kHz, 400 kHz or 1 MHz).

The Intel® Ethernet Connection I219 only runs at a speed of 1250 Mbps, which is 1/2 of the 2.5 Gb/s PCI Express frequency. Each of the PCIe* root ports in the PCH have the ability to run at the 1250-Mbps rate. There is no need to implement a mechanism to detect that the Platform LAN Device is connected. The port configuration (if any), attached to the Platform LAN Device, is pre-loaded from the NVM. The selected port adjusts the transmitter to run at the 1250 Mbps rate and does not need to be PCIe* compliant.

Note: PCIe* validation tools cannot be used for electrical validation of this interface—however, PCIe* layout rules apply for on-board routing.

The integrated GbE controller operates at full-duplex at all supported speeds or half-duplex at 10/100 Mbps. It also adheres to the *IEEE 802.3x Flow Control Specification*.

Note: GbE operation (1000 Mbps) is only supported in S0 mode. In Sx modes, the platform LAN Device may maintain 10/100 Mbps connectivity and use the SMLink interface to communicate with the PCH.

The integrated GbE controller provides a system interface using a PCIe* function. A full memory-mapped or I/O-mapped interface is provided to the software, along with DMA mechanisms for high performance data transfer.

The integrated GbE controller features are:

- Network Features
 - Compliant with the 1 Gb/s Ethernet IEEE 802.3, IEEE 802.3u, IEEE 802.3ab specifications
 - Multi-speed operation: 10/100/1000 Mbps
 - Full-duplex operation at 10/100/1000 Mbps: Half-duplex at 10/100 Mbps
 - Flow control support compliant with the 802.3X specification
 - VLAN support compliant with the 802.3q specification
 - MAC address filters: perfect match unicast filters; multicast hash filtering, broadcast filter and promiscuous mode
 - PCIe*/SMLink interface to GbE PHYs
- Host Interface Features
 - 64 bit address master support for systems using more than 4 GB of physical memory
 - Programmable host memory receive buffers (256 bytes to 16 KB)
 - Intelligent interrupt generation features to enhance driver performance
 - Descriptor ring management hardware for transmit and receive
 - Software controlled reset (resets everything except the configuration space)
 - Message Signaled Interrupts



- Performance Features
 - Configurable receive and transmit data FIFO, programmable in 1 KB increments
 - TCP segmentation off loading features
 - Fragmented UDP checksum off load for packet reassembly
 - IPv4 and IPv6 checksum off load support (receive, transmit, and large send)
 - Split header support to eliminate payload copy from user space to host space
 - Receive Side Scaling (RSS) with two hardware receive queues
 - Supports 9018 bytes of jumbo packets
 - Packet buffer size 32 KB
 - TimeSync off load compliant with IEEE 802.1as specification
 - Platform time synchronization
- Power Management Features
 - Magic Packet* wake-up enable with unique MAC address
 - ACPI register set and power down functionality supporting D0 and D3 states
 - Full wake up support (APM, ACPI)
 - MAC power down at Sx, DM-Off with and without WoL
 - Auto connect battery saver at S0 no link and Sx no link
 - Energy Efficient Ethernet (EEE) support
 - Latency Tolerance Reporting (LTR)
 - ARP and ND proxy support through LAN Connected Device proxy
 - Wake on LAN (WoL) from Deep Sx
 - Windows* InstantGo* Support

21.7.1 GbE PCI Express* Bus Interface

The GbE controller has a PCIe* interface to the host processor and host memory. The following sections detail the bus transactions.

21.7.1.1 Transaction Layer

The upper layer of the host architecture is the transaction layer. The transaction layer connects to the device GbE controller using an implementation specific protocol. Through this GbE controller-to-transaction-layer protocol, the application-specific parts of the device interact with the subsystem and transmit and receive requests to or from the remote agent, respectively.

21.7.1.2 Data Alignment

21.7.1.2.1 4 KB Boundary

PCI requests must never specify an address/length combination that causes a memory space access to cross a 4-KB boundary. It is hardware's responsibility to break requests into 4 KB aligned requests (if needed). This does not pose any requirement on software. However, if software allocates a buffer across a 4 KB boundary, hardware issues multiple requests for the buffer. Software should consider aligning buffers to a 4 KB boundary in cases where it improves performance. The alignment to the 4 KB boundaries is done by the GbE controller. The transaction layer does not do any alignment according to these boundaries.



21.7.1.2.2 PCI Request Size

PCI requests are 128 bytes or less and are aligned to make better use of memory controller resources. Writes, however, can be on any boundary and can cross a 64-byte alignment boundary.

21.7.1.3 Configuration Request Retry Status

The integrated GbE controller might have a delay in initialization due to an NVM read. If the NVM configuration read operation is not completed and the device receives a configuration request, the device responds with a configuration request retry completion status to terminate the request, and thus effectively stalls the configuration request until such time that the sub-system has completed local initialization and is ready to communicate with the host.

21.7.2 Error Events and Error Reporting

21.7.2.1 Completer Abort Error Handling

A received request that violates the LAN Controller programming model will be discarded, for non posted transactions an unsuccessful completion with CA completion status will be returned. For posted transactions if both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#.

21.7.2.2 Unsupported Request Error Handling

A received unsupported request to the LAN Controller will be discarded, for non posted transactions an unsuccessful completion with UR completion status will be returned. The URD bit will be set in ECTL register, If both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#. For posted transactions, if both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#.

21.7.3 Ethernet Interface

The integrated GbE controller provides a complete CSMA/CD function supporting IEEE 802.3 (10 Mbps), IEEE 802.3u (100 Mbps) implementations. It also supports the IEEE 802.3z and IEEE 802.3ab (1000 Mbps) implementations. The device performs all of the functions required for transmission, reception, and collision handling called out in the standards.

The mode used to communicate between the PCH and the Intel® Ethernet Connection I219 supports 10/100/1000 Mbps operation, with both half- and full-duplex operation at 10/100 Mbps, and full-duplex operation at 1000 Mbps.

21.7.3.1 Intel® Ethernet Connection I219

The integrated GbE controller and the Intel® Ethernet Connection I219 communicate through the PCIe* and SMLink0 interfaces. All integrated GbE controller configuration is performed using device control registers mapped into system memory or I/O space. The Platform LAN Phy is configured using the PCI Express or SMLink0 interface.

The integrated GbE controller supports various modes as listed in [Table 21-5](#).



Table 21-5. LAN Mode Support

Mode	System State	Interface Active	Connections
Normal 10/100/1000 Mbps	S0	PCIe*	Intel® Ethernet Connection I219
Manageability and Wake-on-LAN	Sx	SMLink0	Intel® Ethernet Connection I219
Wake-on-LAN	Deep Sx	LAN_WAKE#	Intel® Ethernet Connection I219

21.7.4 PCI Power Management

The integrated GbE controller supports the Advanced Configuration and Power Interface (ACPI) specification as well as Advanced Power Management (APM). This enables the network-related activity (using an internal host wake signal) to wake up the host. For example, from Sx (S3–S5) and Deep Sx to S0.

Note: The Intel® Ethernet Connection I219 must be powered during the Deep Sx state in order to support host wake up from Deep Sx. GPD_2_LAN_WAKE# on the PCH must be configured to support wake from Deep Sx and must be connected to LANWAKE_N on the Platform LAN Connect Device. The SLP_LAN# signal must be driven high (de-asserted) in the Deep Sx state to maintain power to the Platform LAN Connect Device.

The integrated GbE controller contains power management registers for PCI and supports D0 and D3 states. PCIe* transactions are only allowed in the D0 state, except for host accesses to the integrated GbE controller's PCI configuration registers.

Note: SLP_LAN# pin behavior are detailed in [Section 27.7.10.5, "SLP_LAN# Pin Behavior"](#).

§ §



22 Interrupt Interface

22.1 Acronyms

Acronyms	Description
AEOI	Automatic End Of Interrupt
APIC	Advanced Programmable Interrupt Controller
HPET	High Precision Event Timer
PIC	Programmable Interrupt Controller

22.2 Overview

The interrupt controllers are used by the OS to dynamically route PCI interrupts to interrupt requests (IRQs).

22.3 Signal Description

Name	Type	Description
GPP_A6 / SERIRQ / ESPI_CS1#	I/O	Serial Interrupt Request Note: An external Pull-up is required
GPP_A7 / PIRQA# / ESPI_ALERT0#	I/OD	PCI Interrupt Request Ax Note: An external Pull-up is required

22.4 Integrated Pull-Ups and Pull-Downs

None

22.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset*	Immediately after Reset*	S3/S4/S5	Deep Sx
SERIRQ	Primary	Undriven	Undriven	Undriven	OFF
PIRQA#	Primary	Undriven	Undriven	Undriven	OFF
Note: *Reset reference for primary well pins is RSMRST#					



22.6 Functional Description

The PCH supports both APIC and PIC modes.

Interrupt sharing from the perspective of the Interrupt Controller that receives the Interrupts is limited to IRQ 0-23.

- Shareable interrupts requires the Interrupt Controller to track the Assert/De-assert Sideband message from each interrupt source. The Interrupt Controller achieves this through Source ID decode of the message.
- Maintains backwards compatibility with the prior generations where only the lower 24 IRQs are available to support Interrupt Sharing.
- Interrupts are dedicated and not shareable from the perspective of the Interrupt Controller for IRQ 24-119. In other words, not more than 1 Interrupt Initiator is allowed to be assigned to the same IRQ# for IRQ 24-119. For example, GPIO (multi-cause Interrupt Initiator) and Intel® Serial I/O interfaces (I²C, UART, GSPI) (multi-function Interrupt Initiator) should not both generate Assert/De-assert IRQn that maps to IRQ24.
- Possible multi-cause Interrupt Initiator that maps to IRQ24-119 are GPIO, eSPI, and so on.
- Possible multi-function Interrupt Initiators that maps to IRQ24-119 are HD Audio, I²C/UART/GSPI (Intel Serial I/O Interfaces), ISH, and so on.

Interrupt Sharing Compliance Requirement for the Interrupt Initiator are as follows:

1. For multi-cause Initiators (Multiple Interrupt Cause from Single Source and Single Sideband (SB) Port ID, e.g. GPIO, eSPI): If more than 1 interrupt cause has to use the same IRQ#, it has to be aggregated or guaranteed through BIOS/SW to assign a unique IRQ per Interrupt Cause.
2. For multi-function devices (1 Interrupt Cause per Source but many Sources are behind Single SB Port ID, example, Intel® Serial I/O interfaces (I²C, UART, GSPI)): Again if sharing is needed, the interrupts have to be aggregated or guaranteed through SW to ensure a unique IRQ is assigned per Interrupt Cause.
3. IPs that have 1:1 mapping to the IRQ# such as eSPI and LPC are not impacted by this requirement. For eSPI, it is expected that the EC devices aggregate the interrupts before these are communicated to eSPI.
4. Single-cause or Single-function device behind a unique SB Port ID is not subjected to this requirement.

Only level-triggered interrupts can be shared. PCI interrupts (PIRQs) are inherently shared on the board; these should, therefore, be programmed as level-triggered.

The following tables show the mapping of the various interrupts in Non-APIC and APIC modes.

Table 22-1. Interrupt Options - 8259 Mode (Sheet 1 of 2)

IRQ#	Pin	SERIRQ	PCI Message	Internal Modules
0	No	No	No	8254 Counter 0, HPET#0
1	No	Yes	No	Option for configurable sources including GPIO, eSPI and internal PCI/ACPI devices
2	No	No	No	8259 #2 cascade only



Table 22-1. Interrupt Options - 8259 Mode (Sheet 2 of 2)

IRQ#	Pin	SERIRQ	PCI Message	Internal Modules
3:7	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI and internal PCI/ACPI devices
8	No	No	No	RTC, HPET#1
9:10	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal PCI/ACPI devices, SCI and TCO.
11	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal ACPI devices, SCI, TCO, HPET #2
12	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal ACPI devices, HPET#3
13	No	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI devices
14:15	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI and internal ACPI devices
Notes: 1. 8259 Interrupt Request Lines 0, 2 and 8 are non-shareable and dedicated. Only one interrupt source is allowed to use the Interrupt Request Line at any one time. 2. If an interrupt is used for PCI IRQ [A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ). 3. In 8259 mode, PCI interrupts are mapped to IRQ3, 4, 5, 6, 7, 9, 10, 11, 12, 14, or 15. It can be programmed via PIRQ[A-H] Routing Control at ITSS Private CR + Offset 3100h-3107h.				

Table 22-2. Interrupt Options - APIC Mode (Sheet 1 of 2)

IRQ#	Pin	SERIRQ	PCI Message	IRQ Sharable?	Internal Modules
0	No	No	No	No	Cascade from 8259 #1
1	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/PCI devices
2	No	No	No	No	8254 Counter 0, HPET #0 (legacy mode)
3:7	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/PCI devices
8	No	No	No	No	RTC, HPET #1 (legacy mode)
9:10	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/PCI devices, SCI and TCO
11	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/PCI devices, SCI, TCO, HPET #2
12	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/PCI devices, HPET#3
13	No	No	No	Yes	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices
14:15	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices
16	PIRQA	PIRQA	Yes	Yes	Option for configurable sources including internal PIRQA, GPIO, eSPI and internal ACPI/PCI devices



Table 22-2. Interrupt Options - APIC Mode (Sheet 2 of 2)

IRQ#	Pin	SERIRQ	PCI Message	IRQ Sharable?	Internal Modules
17:19	No	PIRQ[B-D]	Yes	Yes	Option for configurable sources including internal PIRQ[B-D], GPIO, eSPI and internal ACPI/PCI devices
20:23	No	No	No	Yes	Option for configurable sources including internal PIRQ[E-H], GPIO, eSPI, SCI, TCO, internal ACPI/PCI devices and HPET
24:119	No	No	No	No	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices
<p>Notes:</p> <ol style="list-style-type: none"> 1. Interrupts 24 through 119 are dedicated and not shareable from the perspective of the Interrupt Controller. Not more than 1 Interrupt source is allowed to be assigned to the same IRQ#. For example, GPIO and Intel® Serial I/O interfaces (I²C, UART, GSPI) should not generate Assert/Deassert_IRQn that maps to IRQ24. Although dedicated, Interrupts 24 through 119 can be configured to be level or edge-triggered. 2. If an interrupt is used for PCI IRQ [A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ). 3. In APIC mode, the PCI interrupts [A:H] are directly mapped to IRQ[16:23]. 4. When programming the polarity of internal interrupt sources on the APIC, interrupts 0 through 15, and 24 through 119 receive active-high internal interrupt sources; interrupts 16 through 23 receive active-low internal interrupt sources. 5. The internal ACPI/PCI devices refer to PCI/PCIe devices configured to the ACPI or PCI function mode. If in ACPI function mode, the device interrupt is map directly to one of the available IRQ. If in PCI function mode, the device interrupt is map to INT[A-D] and then to the IRQ before these devices issue the Interrupt Message using Assert/Deassert_IRQn. 6. PCI Message refers to the downstream Assert/Deassert_INT[A-D] messages forwarded from the processor complex. 					

The following signals are associated with the Interrupt Logic.

Table 22-3. Interrupt Logic Signals

Signal Name	C3	S3	S5
SERIRQ	Can be running	Off	Off
PIRQA#	Can go active	Off	Off



22.6.1 8259 Interrupt Controllers (PIC)

The ISA-compatible interrupt controller (PIC) incorporates the functionality of two 8259 interrupt controllers. The following table shows how the cores are connected.

Table 22-4. Interrupt Controllers PIC

8259	8259 Input	Typical Interrupt Source	Connected Pin/Function
Master	0	Internal	Internal Timer/Counter 0 output or Multimedia Timer #0
	1	Keyboard	IRQ1 via SERIRQ. Option for configurable sources including eSPI, GPIO, internal ACPI devices.
	2	Internal	Slave Controller INTR output
	3	Serial Port A	IRQ3 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	4	Serial Port B	IRQ4 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	5	Parallel Port/Generic	IRQ5 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	6	Floppy Disk	IRQ6 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	7	Parallel Port/Generic	IRQ7 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
Slave	0	Real Time Clock	Inverted IRQ8# from internal RTC or Multimedia Timer #1
	1	Generic	IRQ9 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO.
	2	Generic	IRQ10 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO.
	3	Generic	IRQ11 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO or HPET #2.
	4	PS/2 Mouse	IRQ12 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO or HPET #3.
	5	Internal	IRQ13 from configurable sources including PIRQx, eSPI, GPIO, internal ACPI devices.
	6	Internal	IRQ14 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	7	Internal	IRQ15 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.

The slave controller is cascaded onto the master controller through master controller interrupt input 2. This means there are only 15 possible interrupts for PCH PIC.

Interrupts can individually be programmed to be edge or level triggered, except for IRQ0, IRQ1, IRQ2 and IRQ8# which always default to edge.

Active-low interrupt sources, such as the PIRQ#s, are internally inverted before being sent to the PIC. In the following descriptions of the 8259s, the interrupt levels are in reference to the signals at the internal interface of the 8259s, after the required inversions have occurred. Therefore, the term "high" indicates "active", which means "low" on an originating PIRQ#.



22.6.2 Interrupt Handling

22.6.2.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 22-5 defines the IRR, ISR, and IMR.

Table 22-5. Interrupt Status Registers

Bit	Description
IRR	Interrupt Request Register. This bit is set on a low to high transition of the interrupt line in edge mode, and by an active high level in level mode. This bit is set whether or not the interrupt is masked. However, a masked interrupt will not generate INTR.
ISR	Interrupt Service Register. This bit is set, and the corresponding IRR bit cleared, when an interrupt acknowledge cycle is seen, and the vector returned is for that interrupt.
IMR	Interrupt Mask Register. This bit determines whether an interrupt is masked. Masked interrupts will not generate INTR.

22.6.2.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle that is translated by the host bridge into a PCI Interrupt Acknowledge Cycle to the PCH. The PIC translates this command into two internal INTA# pulses expected by the 8259 cores. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave sends the interrupt vector to the processor with the acknowledged interrupt code. This code is based on Bits [7:3] of the corresponding ICW2 register, combined with three bits representing the interrupt within that controller.

Table 22-6. Content of Interrupt Vector Byte

Master, Slave Interrupt	Bits [7:3]	Bits [2:0]
IRQ7,15	ICW2[7:3]	111
IRQ6,14		110
IRQ5,13		101
IRQ4,12		100
IRQ3,11		011
IRQ2,10		010
IRQ1,9		001
IRQ0,8		000

22.6.2.3 Hardware/Software Interrupt Sequence

1. One or more of the Interrupt Request lines (IRQ) are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor if an asserted interrupt is not masked.
3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle. The cycle is translated into a PCI interrupt acknowledge cycle by the host bridge. This command is broadcast over PCI by the PCH.
4. Upon observing its own interrupt acknowledge cycle on PCI, the PCH converts it into the two cycles that the internal 8259 pair can respond to. Each cycle appears

as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.

5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine if it must respond with an interrupt vector during the second INTA# pulse.
6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. If no interrupt request is present because the request was too short in duration, the PIC returns vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

22.6.3 Initialization Command Words (ICWx)

Before operation can begin, each 8259 must be initialized. In the PCH, this is a four byte sequence. The four initialization command words are referred to by their acronyms: ICW1, ICW2, ICW3, and ICW4.

The base address for each 8259 initialization command word is a fixed location in the I/O memory space: 20h for the master controller, and A0h for the slave controller.

22.6.3.1 ICW1

An I/O write to the master or slave controller base address with data bit 4 equal to 1 is interpreted as a write to ICW1. Upon sensing this write, the PCH's PIC expects three more byte writes to 21h for the master controller, or A1h for the slave controller, to complete the ICW sequence.

A write to ICW1 starts the initialization sequence during which the following automatically occur:

1. Following initialization, an interrupt request (IRQ) input must make a low-to-high transition to generate an interrupt.
2. The Interrupt Mask Register is cleared.
3. IRQ7 input is assigned priority 7.
4. The slave mode address is set to 7.
5. Special mask mode is cleared and Status Read is set to IRR.

22.6.3.2 ICW2

The second write in the sequence (ICW2) is programmed to provide bits [7:3] of the interrupt vector that will be released during an interrupt acknowledge. A different base is selected for each interrupt controller.

22.6.3.3 ICW3

The third write in the sequence (ICW3) has a different meaning for each controller.



- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. Within the PCH, IRQ2 is used. Therefore, Bit 2 of ICW3 on the master controller is set to a 1, and the other bits are set to 0s.
- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller if the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and if it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

22.6.3.4 ICW4

The final write in the sequence (ICW4) must be programmed for both controllers. At the very least, Bit 0 must be set to a 1 to indicate that the controllers are operating in an Intel Architecture-based system.

22.6.4 Operation Command Words (OCW)

These command words reprogram the interrupt controller to operate in various interrupt modes.

- OCW1 masks and unmasks interrupt lines.
- OCW2 controls the rotation of interrupt priorities when in rotating priority mode, and controls the EOI function.
- OCW3 sets up ISR/IRR reads, enables/disables the special mask mode (SMM), and enables/disables polled interrupt mode.

22.6.5 Modes of Operation

22.6.5.1 Fully-Nested Mode

In this mode, interrupt requests are ordered in priority from 0 through 7, with 0 being the highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the ISR for the interrupt is set. This ISR bit remains set until: the processor issues an EOI command immediately before returning from the service routine; or if in AEOI mode, on the trailing edge of the second INTA#. While the ISR bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels generate another interrupt. Interrupt priorities can be changed in the rotating priority mode.

22.6.5.2 Special Fully-Nested Mode

This mode is used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully-nested mode is programmed to the master controller. This mode is similar to the fully-nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave are recognized by the master and initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.

- When exiting the Interrupt Service Routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a Non-Specific EOI command to the slave and then reading its ISR. If it is 0, a Non-Specific EOI can also be sent to the master.

22.6.5.3 Automatic Rotation Mode (Equal Priority Devices)

In some applications, there are a number of interrupting devices of equal priority. Automatic rotation mode provides for a sequential 8-way rotation. In this mode, a device receives the lowest priority after being serviced. In the worst case, a device requesting an interrupt has to wait until each of seven other devices are serviced at most once.

There are two ways to accomplish automatic rotation using OCW2: the Rotation on Non-Specific EOI Command (R=1, SL=0, EOI=1) and the rotate in automatic EOI mode which is set by (R=1, SL=0, EOI=0).

22.6.5.4 Specific Rotation Mode (Specific Priority)

Software can change interrupt priorities by programming the bottom priority. For example, if IRQ5 is programmed as the bottom priority device, then IRQ6 is the highest priority device. The Set Priority Command is issued in OCW2 to accomplish this, where: R=1, SL=1, and LO-L2 is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes can be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (R=1, SL=1, EOI=1 and LO-L2=IRQ level to receive bottom priority).

22.6.5.5 Poll Mode

Poll mode can be used to conserve space in the interrupt vector table. Multiple interrupts that can be serviced by one Interrupt Service Routine do not need separate vectors if the service routine uses the poll command. Poll mode can also be used to expand the number of interrupts. The polling Interrupt Service Routine can call the appropriate service routine, instead of providing the interrupt vectors in the vector table. In this mode, the INTR output is not used and the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting P=1 in OCW3. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit if there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read contains a 1 in Bit 7 if there is an interrupt, and the binary code of the highest priority level in Bits 2:0.

22.6.5.6 Edge and Level Triggered Mode

In ISA systems this mode is programmed using Bit 3 in ICW1, which sets level or edge for the entire controller. In the PCH, this bit is disabled and a register for edge and level triggered mode selection, per interrupt input, is included. This is the Edge/Level control Registers ELCR1 and ELCR2.

If an ELCR bit is 0, an interrupt request will be recognized by a low-to-high transition on the corresponding IRQ input. The IRQ input can remain high without generating another interrupt. If an ELCR bit is 1, an interrupt request will be recognized by a high



level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

In both the edge and level triggered modes, the IRQ inputs must remain active until after the falling edge of the first internal INTA#. If the IRQ input goes inactive before this time, a default IRQ7 vector is returned.

22.6.5.7 End Of Interrupt (EOI) Operations

An EOI can occur in one of two fashions: by a command word write issued to the PIC before returning from a service routine, the EOI command; or automatically when AEOI bit in ICW4 is set to 1.

22.6.5.8 Normal End of Interrupt

In normal EOI, software writes an EOI command before leaving the Interrupt Service Routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific. When a Non-Specific EOI command is issued, the PIC clears the highest ISR bit of those that are set to 1. Non-Specific EOI is the normal mode of operation of the PIC within the PCH, as the interrupt being serviced currently is the interrupt entered with the interrupt acknowledge. When the PIC is operated in modes that preserve the fully nested structure, software can determine which ISR bit to clear by issuing a Specific EOI. An ISR bit that is masked is not cleared by a Non-Specific EOI if the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

22.6.5.9 Automatic End of Interrupt Mode

In this mode, the PIC automatically performs a Non-Specific EOI operation at the trailing edge of the last interrupt acknowledge pulse. From a system standpoint, this mode should be used only when a nested multi-level interrupt structure is not required within a single PIC. The AEOI mode can only be used in the master controller and not the slave controller.

22.6.6 Masking Interrupts

22.6.6.1 Masking on an Individual Interrupt Request

Each interrupt request can be masked individually by the Interrupt Mask Register (IMR). This register is programmed through OCW1. Each bit in the IMR masks one interrupt channel. Masking IRQ2 on the master controller masks all requests for service from the slave controller.

22.6.6.2 Special Mask Mode

Some applications may require an Interrupt Service Routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The special mask mode enables all interrupts not masked by a bit set in the Mask Register. Normally, when an Interrupt Service Routine acknowledges an interrupt without issuing an EOI to clear the ISR bit, the interrupt controller inhibits all lower priority requests. In the special mask mode, any interrupts may be selectively enabled



by loading the Mask Register with the appropriate pattern. The special Mask Mode is set by OCW3.SSMM and OCW3.SMM set, and cleared when OCW3.SSMM and OCW3.SMM are cleared.

22.6.7 Steering PCI Interrupts

The PCH can be programmed to allow PIRQ[A:D]# to be internally routed to interrupts 3-7, 9-12, 14 or 15, through the PARC, PBRC, PCRC, PDRC, PERC, PFRC, PGRC, and PHRC registers in the chipset configuration section. One or more PIRQx# lines can be routed to the same IRQx input.

The PIRQx# lines are defined as active low, level sensitive. When PIRQx# is routed to specified IRQ line, software must change the corresponding ELCR1 or ELCR2 register to level sensitive mode. The PCH will internally invert the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an ISA device.

22.7 Advanced Programmable Interrupt Controller (APIC) (D31:F0)

In addition to the standard ISA-compatible PIC described in the previous section, the PCH incorporates the APIC. While the standard interrupt controller is intended for use in a uni-processor system, APIC can be used in either a uni-processor or multi-processor system.

22.7.1 Interrupt Handling

The I/O APIC handles interrupts very differently than the 8259. Briefly, these differences are:

- **Method of Interrupt Transmission.** The I/O APIC transmits interrupts through memory writes on the normal data path to the processor, and interrupts are handled without the need for the processor to run an interrupt acknowledge cycle.
- **Interrupt Priority.** The priority of interrupts in the I/O APIC is independent of the interrupt number. For example, interrupt 10 can be given a higher priority than interrupt 3.
- **More Interrupts.** The I/O APIC in the PCH supports a total of 24 interrupts.
- **Multiple Interrupt Controllers.** The I/O APIC architecture allows for multiple I/O APIC devices in the system with their own interrupt vectors.

22.7.2 Interrupt Mapping

The I/O APIC within the PCH supports 40 APIC interrupts. Each interrupt has its own unique vector assigned by software.

22.7.3 PCI/PCI Express* Message-Based Interrupts

When external devices through PCI/PCI Express* wish to generate an interrupt, they will send the message defined in the *PCI Express* Base Specification*, Revision 2.0 for generating INTA# – INTD#. These will be translated internal assertions/de-assertions of INTA# – INTD#.



22.7.4 IOxAPIC Address Remapping

To support Intel® Virtualization Technology (Intel® VT), interrupt messages are required to go through similar address remapping as any other memory request. Address remapping allows for domain isolation for interrupts, so a device assigned in one domain is not allowed to generate an interrupt to another domain.

The address remapping is based on the Bus: Device: Function field associated with the requests. The internal APIC is required to initiate the interrupt message using a unique Bus: Device: Function.

The PCH allows BIOS to program the unique Bus: Device: Function address for the internal APIC. This address field does not change the APIC functionality and the APIC is not promoted as a stand-alone PCI device. Refer Device 31: Function 0 Offset 6Ch for additional information.

22.7.5 External Interrupt Controller Support

The PCH supports external APICs off of PCI Express ports but does not support APICs on the PCI bus. The EOI special cycle is only forwarded to PCI Express ports.

22.8 Serial Interrupt

The PCH supports a serial IRQ scheme. This allows a single signal to be used to report interrupt requests. The signal used to transmit this information is shared between the PCH and all participating peripherals. The signal line, SERIRQ, is synchronous to 24 MHz CLKOUT_LPC, and follows the sustained tri-state protocol that is used by all PCI signals. This means that if a device has driven SERIRQ low, it will first drive it high synchronous to PCI clock and release it the following PCI clock. The serial IRQ protocol defines this sustained tri-state signaling in the following fashion:

- **S – Sample Phase**, Signal driven low
- **R – Recovery Phase**, Signal driven high
- **T – Turn-around Phase**, Signal released

The PCH supports a message for 21 serial interrupts. These represent the 15 ISA interrupts (IRQ0–1, 3–15), the four PCI interrupts, and the control signals SMI# and IOCHK#. The serial IRQ protocol does not support the additional APIC interrupts (20–23).

Note: IRQ14 and IRQ15 are special interrupts and maybe used by the GPIO controller when it is running GPIO driver mode. When the GPIO controller operates in GPIO driver mode, IRQ14 and IRQ15 shall not be utilized by the SERIRQ stream nor mapped to other interrupt sources, and instead come from the GPIO controller. If the GPIO controller is entirely in ACPI mode, these interrupts can be mapped to other devices accordingly.



22.8.1 Start Frame

The serial IRQ protocol has two modes of operation which affect the start frame. These two modes are: Continuous, where the PCH is solely responsible for generating the start frame; and Quiet, where a serial IRQ peripheral is responsible for beginning the start frame.

The mode that must first be entered when enabling the serial IRQ protocol is continuous mode. In this mode, the PCH asserts the start frame. This start frame is 4, 6, or 8 PCI clocks wide based upon the Serial IRQ Control Register, bits 1:0 at 64h in D31:F0 configuration space. This is a polling mode.

When the serial IRQ stream enters quiet mode (signaled in the Stop Frame), the SERIRQ line remains inactive and pulled up between the Stop and Start Frame until a peripheral drives the SERIRQ signal low. The PCH senses the line low and continues to drive it low for the remainder of the Start Frame. Since the first PCI clock of the start frame was driven by the peripheral in this mode, the PCH drives the SERIRQ line low for 1 PCI clock less than in continuous mode. This mode of operation allows for a quiet, and therefore lower power, operation.

Once the Start frame has been initiated, all of the SERIRQ peripherals must start counting frames based on the rising edge of SERIRQ. Each of the IRQ/DATA frames has exactly 3 phases of 1 clock each:

- **Sample Phase**—During this phase, the SERIRQ device drives SERIRQ low if the corresponding interrupt signal is low. If the corresponding interrupt is high, then the SERIRQ devices tri-state the SERIRQ signal. The SERIRQ line remains high due to Pull-up resistors (there is no internal Pull-up resistor on this signal, an external Pull-up resistor is required). A low level during the IRQ0–1 and IRQ2–15 frames indicates that an active-high ISA interrupt is not being requested, but a low level during the PCI INT[A:D], SMI#, and IOCHK# frame indicates that an active-low interrupt is being requested.
- **Recovery Phase**—During this phase, the device drives the SERIRQ line high if in the Sample Phase it was driven low. If it was not driven in the sample phase, it is tri-stated in this phase.
- **Turn-around Phase**—The device tri-states the SERIRQ line.

22.8.2 Stop Frame

After all data frames, a Stop Frame is driven by the PCH. The SERIRQ signal is driven low by the PCH for 2 or 3 PCI clocks. The number of clocks is determined by the SERIRQ configuration register. The number of clocks determines the next mode.

Table 22-7. Stop Frame Explanation

Stop Frame Width	Next Mode
2 PCI clocks	Quiet Mode. Any SERIRQ device may initiate a Start Frame
3 PCI clocks	Continuous Mode. Only the host (the PCH) may initiate a Start Frame



22.8.3 Specific Interrupts Not Supported Using SERIRQ

There are three interrupts seen through the serial stream that are not supported by the PCH. These interrupts are generated internally, and are not sharable with other devices within the system. These interrupts are:

- IRQ0. Heartbeat interrupt generated off of the internal 8254 counter 0.
- IRQ8#. RTC interrupt can only be generated internally.
- IRQ13. Reserved internally.

The PCH ignores the state of these interrupts in the serial stream, and does not adjust their level based on the level seen in the serial stream. Data Frame Format.

Table 22-8 shows the format of the data frames. For the PCI interrupts (A–D), the output from the PCH is AND’d with the PCI input signal. This way, the interrupt can be signaled using both the PCI interrupt input signal and using the SERIRQ signal (they are shared).

Table 22-8. Data Frame Format

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
1	IRQ0	2	Ignored. IRQ0 can only be generated using the internal 8524
2	IRQ1	5	Before port 60h latch
3	SMI#	8	Causes SMI# if low. Will set the SERIRQ_SMI_STS bit.
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	
8	IRQ7	23	
9	IRQ8	26	Ignored. IRQ8# can only be generated internally.
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	Before port 60h latch
14	IRQ13	41	Ignored.
15	IRQ14	44	Not attached to GPIO logic
16	IRQ15	47	Not attached to GPIO logic
17	IOCHCK#	50	Same as ISA IOCHCK# going active
18	PCI INTA#	53	Drive PIRQA#
19	PCI INTB#	56	Drive PIRQB#
20	PCI INTC#	59	Drive PIRQC#
21	PCI INTD#	62	Drive PIRQD#

§ §

23 Integrated Sensor Hub (ISH)

23.1 Acronyms

Acronyms	Description
Intel® CSME	Intel® Converged Security and Management Engine
I ² C	Inter-Integrated Circuit
IPC	Inter Process Communication
SPI	Serial Peripheral Interface
ISH	Integrated Sensor Hub
PMU	Power Management Unit
SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver/Transmitter

23.2 References

Specification	Location
I ² C Specification Version 6.0	http://www.nxp.com/docs/en/user-guide/UM10204.pdf

23.3 Overview

The Integrated Sensor Hub (ISH) serves as the connection point for many of the sensors on a platform. The ISH is designed with the goal of “Always On, Always Sensing” and it provides the following functions to support this goal:

- Acquisition/sampling of sensor data.
- The ability to combine data from individual sensors to create a more complex virtual sensor that can be directly used by the firmware/OS.
- Low power operation through clock and power gating of the ISH blocks together with the ability to manage the power state of the external sensors.
- The ability to operate independently when the host platform is in a low power state (S0ix only).
- Ability to provide sensor-related data to other subsystems within the PCH, such as the Intel® CSME.

The ISH consists of the following key components:

- A combined cache for instructions and data.
 - ROM space intended for the bootloader.
 - SRAM space for code and data.
- Interfaces to sensor peripherals (I²C, UART, SPI, GPIO).
- An interface to main memory.
- Out of Band signals for clock and wake-up control.
- Inter Process Communications to the Host and Intel® CSME.
- Part of the PCI tree on the host.



23.4 Signal Description

Name	Type	Description
ISH_I2C0_SDA/GPP_H19	I/OD	ISH I ² C 0 Data
ISH_I2C0_SCL/GPP_H20	I/OD	ISH I ² C 0 Clk
ISH_I2C1_SDA/GPP_H21	I/OD	ISH I ² C 1 Data
ISH_I2C1_SCL/GPP_H22	I/OD	ISH I ² C 1 Clk
ISH_I2C2_SDA / GPP_D4 / I2C3_SDA / SBK4 / BK4	I/OD	ISH I ² C 2 Data
ISH_I2C2_SCL / GPP_D23 / I2C3_SCL	I/OD	ISH I ² C 2 Clk
ISH_GP0/GPP_A18	I/O	ISH GPIO 0
ISH_GP1/GPP_A19	I/O	ISH GPIO 1
ISH_GP2/GPP_A20	I/O	ISH GPIO 2
ISH_GP3/GPP_A21	I/O	ISH GPIO 3
ISH_GP4/GPP_A22	I/O	ISH GPIO 4
ISH_GP5/GPP_A23	I/O	ISH GPIO 5
ISH_GP6 / GPP_A12 / BM_BUSY# / SX_EXIT_HOLDOFF#	I/O	ISH GPIO 6
ISH_GP7/ GPP_A17 / SD_VDD1_PWR_EN#	I/O	ISH GPIO 7
ISH_UART0_TXD / GPP_D14 / I2C2_SCL	O	ISH UART 0 Transmit Data
ISH_UART0_RXD /GPP_D13 / I2C2_SDA	I	ISH UART 0 Receive Data
ISH_UART0_RTS# /GPP_D15 / GSPI2_CS1# / CNV_WFEN	O	ISH UART 0 Request To Send
ISH_UART0_CTS# /GPP_D16 / CNV_WCEN	I	ISH UART 0 Clear to Send
ISH_UART1_TXD /GPP_C13 / UART1_TXD	O	ISH UART 1 Transmit Data
ISH_UART1_RXD /GPP_C12 / UART1_RXD	I	ISH UART 1 Receive Data
ISH_UART1_RTS# /GPP_C14 / UART1_RTS#	O	ISH UART 1 Request To Send
ISH_UART1_CTS# / GPP_C15 / UART1_CTS#	I	ISH UART 1 Clear to Send
ISH_SPI_CS# / GPP_D9 / GSPI2_CS0#	O	ISH SPI 2 Chip Select
ISH_SPI_CLK / GPP_D10 / GSPI2_CLK	O	ISH SPI 2 Clock
ISH_SPI_MISO / GPP_D11 / GSPI2_MISO / GP_BSSB_CLK	I	ISH SPI 2 MISO
ISH_SPI_MOSI / GPP_D12 / GSPI2_MOSI / GP_BSSB_DI	O	ISH SPI 2 MOSI Note: This signal is also utilized as a strap. Refer the pin strap chapter for more info.

23.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
ISH_SPI_MISO	Pull-Down	20 kohm ± 30%	



23.6 I/O Signal Planes and States

Table 23-1. I/O Signal Planes and States

Signal Name	Power Plane	During Reset*	Immediately after Reset*	S3/S4/S5	Deep Sx
ISH_I2C0_SDA	Primary	Undriven	Undriven	Undriven	OFF
ISH_I2C0_SCL	Primary	Undriven	Undriven	Undriven	OFF
ISH_I2C1_SDA	Primary	Undriven	Undriven	Undriven	OFF
ISH_I2C1_SCL	Primary	Undriven	Undriven	Undriven	OFF
ISH_I2C2_SDA	Primary	Undriven	Undriven	Undriven	OFF
ISH_I2C2_SCL	Primary	Undriven	Undriven	Undriven	OFF
ISH_GP[7:0]	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART0_TXD	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART0_RXD	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART0_RTS#	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART0_CTS#	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART1_TXD	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART1_RXD	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART1_RTS#	Primary	Undriven	Undriven	Undriven	OFF
ISH_UART1_CTS#	Primary	Undriven	Undriven	Undriven	OFF
ISH_SPI_CS#	Primary	Undriven	Undriven	Undriven	OFF
ISH_SPI_CLK	Primary	Undriven	Undriven	Undriven	OFF
ISH_SPI_MISO	Primary	Undriven	Undriven	Undriven	OFF
ISH_SPI_MOSI	Primary	Undriven	Undriven	Undriven	OFF
Note: *Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.					

23.7 Functional Description

23.7.1 ISH Micro-Controller

The ISH is operated by a micro-controller. This core provides localized sensor aggregation and data processing, thus off loading the processor and lowering overall platform average power. The core supports an in-built local APIC that receives messages from the IOAPIC. A local boot ROM with FW for initialization is also part of the core.

23.7.2 SRAM

The local SRAM is used for ISH FW code storage and to read/write operational data. The local SRAM block includes both the physical SRAM as well as the controller logic. The SRAM is a total of 640 kbytes organized into banks of 32 kB each and is 32-bit wide. The SRAM is shared with Intel® CSME as shareable memory. To protect against memory errors, the SRAM includes ECC support. The ECC mechanism is able to detect multi-bit errors and correct for single bit errors. The ISH firmware has the ability to put unused SRAM banks into lower power states to reduce power consumption.



23.7.3 PCI Host Interface

The ISH provides access to PCI configuration space via a PCI Bridge. Type 0 Configuration Cycles from the host are directed to the PCI configuration space.

23.7.3.1 MMIO Space

A memory-mapped Base Address Register (BAR0) with a set of functional memory-mapped registers is accessible to the host via the Bridge. These registers are owned by the driver running on the Host OS.

The bridge also supports a second BAR (BAR1) that is an alias of the PCI Configuration space. It is used only in ACPI mode (that is, when the PCI configuration space is hidden).

23.7.3.2 DMA Controller

The DMA controller supports up to 64-bit addressing.

23.7.3.3 PCI Interrupts

The PCI bridge supports standard PCI interrupts, delivered using IRQx to the system IOAPIC and not using an MSI to the host CPU.

23.7.3.4 PCI Power Management

PME is not supported in ISH.

23.7.4 Power Domains and Management

23.7.4.1 ISH Power Management

The various functional blocks within the ISH are all on the primary power plane within the PCH. The ISH is only intended for use during S0 and S0ix states. There is no support for operation in S3, S4, or S5 states. Thus, the system designer must ensure that the inputs to the ISH signals are not driven high while the PCH is in S3–S5 state.

The unused banks of the ISH SRAM can be power-gated by the ISH Firmware.

23.7.4.2 External Sensor Power Management

External sensors can generally be put into a low power state through commands issued over the I/O interface (I²C). Refer to the datasheets of the individual sensors to obtain the commands to be sent to the peripheral.

23.7.5 ISH IPC

The ISH has IPC channels for communication with the Host Processor and Intel® CSME. The functions supported by the ISH IPC block are listed below.

Function 1: Allows for messages and interrupts to be sent from an initiator (such as the ISH) and a target (such as the Intel® CSME). The supported initiator -> target flows using this mechanism are shown in the table below



Table 23-2. IPC Initiator -> Target flows

Initiator	Target
ISH	Host processor
Host processor	ISH
ISH	Intel® CSME
Intel® CSME	ISH

Function 2: Provides status registers and remap registers that assist in the boot flow and debug. These are simple registers with dual access read/write support and cause no interrupts.

23.7.6 ISH Interrupt Handling via IOAPIC (Interrupt Controller)

The PCH legacy IOAPIC is the interrupt controller for the ISH. It collects inputs from various internal blocks and sends interrupt messages to the ISH controller. When there is a change on one of its inputs, the IOAPIC sends an interrupt message to the ISH controller.

The PCH IOAPIC allows each interrupt input to be active high or active low and edge or level triggered.

23.7.7 ISH I²C Controllers

The ISH supports three I²C controllers capable of operating at speeds up to 2.4 Mbps each. The I²C controllers are completely independent of each other: they do not share any pins, memory spaces, or interrupts.

The ISH's I²C host controllers share the same general specifications:

- Master Mode Only (all peripherals must be slave devices)
- Support for the following operating speeds:
 - Standard mode: 100 kbps
 - Fast Mode: 400 kbps
 - Fast Mode Plus: 1 000 kbps
 - High Speed Mode: 2400 kbps
- Support for both 7-bit and 10-bit addressing formats on the I²C bus
- FIFO of 64 bytes with programmable watermarks/thresholds

23.7.8 ISH UART Controller

The ISH has two UART ports, each comprised of a four-wire, bi-directional point-to-point connection between the ISH and a peripheral.

The UART has the following Capabilities:

- Support for operating speeds up to 4 Mbps
- Support for auto flow control using the RTS#/CTS# signals
- 64-byte FIFO



- DMA support to allow direct transfer to the ISH local SRAM without intervention by the controller. This saves interrupts on packets that are longer than the FIFO or when there are back-to-back packets to send or receive.

23.7.9 ISH GSPI Controller

The ISH supports one SPI controller comprises of four-wired interface connecting the ISH to external sensor devices.

The SPI controller includes:

- Master Mode Only
- Single Chip Select
- Half Duplex operation only
- Programmable SPI clock frequency range with maximum rate of 24 Mbits/sec
- FIFO of 64 bytes with programmable thresholds
- Support Programmable character length (2 to 16 bits)

23.7.10 ISH GPIOs

The ISH supports eight dedicated GPIOs.

§ §

24 Low Pin Count (LPC)

24.1 Acronyms

Acronyms	Description
LPC	Low Pin Count

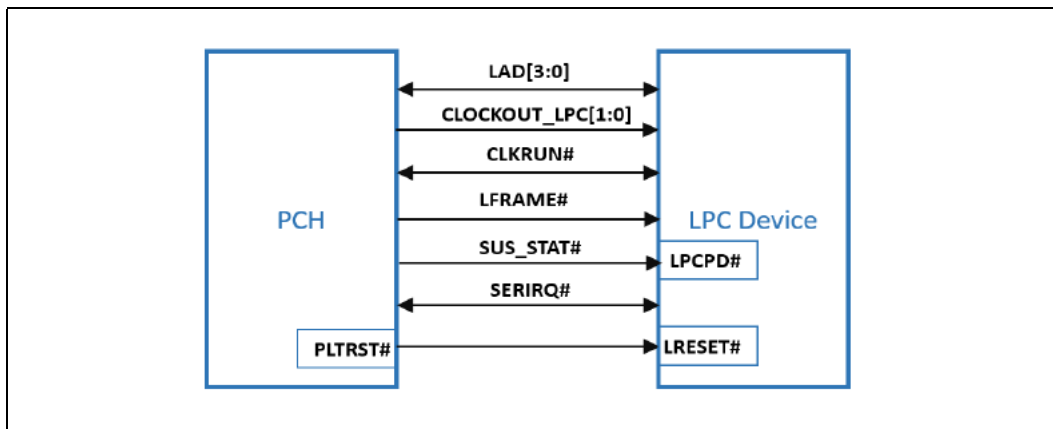
24.2 References

Specification	Location
Intel® Low Pin Count Interface Specification Revision 1.1	https://www.intel.com/content/dam/www/program/design/us/en/documents/low-pin-count-interface-specification.pdf

24.3 Overview

The PCH implements an LPC interface as described in the *Low Pin Count Interface Specification, Revision 1.1*. The LPC interface to the PCH is shown in the following figure.

Figure 24-1. LPC Interface Diagram



24.4 Signal Description

Name	Type	Description
GPP_A1 / LAD0 / ESPI_IO0	I/O	LPC Multiplexed Command, Address, Data. For LAD0, internal Pull-up is provided.
GPP_A2 / LAD1 / ESPI_IO1	I/O	LPC Multiplexed Command, Address, Data. For LAD1, internal Pull-up is provided.
GPP_A3 / LAD2 / ESPI_IO2	I/O	LPC Multiplexed Command, Address, Data. For LAD2, internal Pull-up is provided.



Name	Type	Description
GPP_A4 / LAD3 / ESPI_IO3	I/O	LPC Multiplexed Command, Address, Data. For LAD3, internal Pull-up is provided.
GPP_A5 / LFRAME# / ESPI_CS#	O	LPC Frame: LFRAME# indicates the start of an LPC cycle, or an abort.
GPP_A9 / CLKOUT_LPC0 / ESPI_CLK	O	Low Pin Count (LPC) Clock Outputs: Single-Ended 24-MHz output to various single load connectors/ devices.
GPP_A10 / CLKOUT_LPC1	O	Low Pin Count (LPC) Clock Outputs: Single-Ended 24-MHz output to various single load connectors/ devices.
GPP_A8 / CLKRUN#	I/O	LPC Clock Run for control of CLKOUT_LPC[1:0]: Connects to peripherals that need to request clock restart or prevention of clock stopping.
GPP_A6 / SERIRQ / ESPI_CS1#	I/O	This signal implements the serial interrupt protocol. Note: An external Pull-up to V3.3S power rail is required.
GPP_A14 / SUS_STAT# / ESPI_RESET#	O	LPC Mode - Suspend Status: This signal is asserted by the PCH to indicate that the system will be entering a low power state soon. This can be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It can also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes.

24.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
LAD[3:0]	Pull-up	15 - 40 kohm	

24.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset*	Immediately after Reset*	S3/S4/S5	Deep Sx
LAD[3:0]	Primary	Internal Pull-up	Internal Pull-up	Undriven	Off
LFRAME#	Primary	Driven High	Driven High	Driven Low	Off
CLKOUT_LPC0	Primary	Toggling	Toggling	Driven Low	Off
CLKOUT_LPC1	Primary	Toggling	Toggling	Driven Low	Off
CLKRUN#	Primary	Undriven	Undriven	Undriven	Off
SERIRQ	Primary	Undriven	Undriven	Undriven	Off
SUS_STAT#	Primary	Driven Low	Driven Low	Driven Low	Off

Note: *Reset reference for primary well pins is RSMRST#

24.7 Functional Description

The PCH LPC interface supports the *Low Pin Count Interface Specification*. The bus operates at 24 MHz clock frequency.

24.7.1 LPC Cycle Types

The PCH implements the cycle types shown in [Table 24-1](#).



Table 24-1. LPC Cycle Types Supported

Cycle Type	Comment
Memory Read	1 byte only—(Refer Note 1 below)
Memory Write	1 byte only—(Refer Note 1 below)
I/O Read	1 byte only—The PCH breaks up 16-bit and 32-bit processor cycles into multiple 8-bit transfers.
I/O Write	1 byte only—The PCH breaks up 16-bit and 32-bit processor cycles into multiple 8-bit transfers.
Firmware Memory Read	1 byte only
Firmware Memory Write	1 byte only
Notes: 1. The PCH provides a single generic memory range (LGMR) for decoding memory cycles and forwarding them as LPC Memory cycles on the LPC bus. The LGMR memory decode range is 64 KB in size, and can be defined as being anywhere in the 4 GB memory space. This range needs to be configured by BIOS during POST to provide the necessary memory resources. BIOS should advertise the LPC Generic Memory Range as Reserved to the OS in order to avoid resource conflict. For larger transfers, the PCH performs multiple 8-bit transfers. If the cycle is not claimed by any peripheral, it is subsequently aborted, and the PCH returns a value of all 1s to the processor. This is done to maintain compatibility with ISA memory cycles where pull-up resistors would keep the bus high if no device responds.	

24.7.2 Start Field Definition

Table 24-2. Start Field Bit Definitions

Bits[3:0] Encoding	Definition
0000	Start of cycle for a generic target
1111	Stop/Abort: End of a cycle for a target.
Note: All other encodings are RESERVED.	

24.7.3 Cycle Type/Direction (CYCTYPE + DIR)

The PCH always drives bit0 of this field to 0. Table 24-3 shows the valid bit encodings.

Table 24-3. Cycle Type Bit Definitions

Bits[3:2]	Bit1	Definition
00	0	I/O Read
00	1	I/O Write
01	0	Memory Read
01	1	Memory Read
11	x	Reserved. If a peripheral performing a bus master cycle generates this value, the PCH aborts the cycle.
Note: All other encodings are RESERVED.		



24.7.4 Size

Bits[3:2] are reserved. The PCH always drives them to 00. Bits[1:0] are encoded as listed in Table 24-4.

Table 24-4. Transfer Size Bit Definition

Bits[1:0]	Size
00	8-bit transfer (1 byte)
01	16-bit transfer (2 bytes)
10	Reserved
11	32-bit transfer (4 bytes)

24.7.4.1 SYNC

Valid values for the SYNC field are shown in Table 24-5.

24.7.5 SYNC Timeout

Table 24-5. SYNC Bit Definition

Bits[3:0]	Indication
0000	Ready: SYNC achieved with no error.
0101	Short Wait: Part indicating wait-states. For bus master cycles, the PCH does not use this encoding. Instead, the PCH uses the long wait encoding.
0110	Long Wait: Part indicating wait-states, and many wait-states will be added. This encoding driven by the PCH for bus master cycles, rather than the Short Wait (0101).
1010	Error: Sync achieved with error. This is generally used to replace the SERR# or IOCHK# signal on the PCI/ISA bus. It indicates that the data is to be transferred, but there is a serious error in this transfer.
Notes: <ol style="list-style-type: none"> All other combinations are RESERVED. If the LPC controller receives any SYNC returned from the device other than short (0101), long wait (0110), or ready (0000) when running a FWH cycle, indeterminate results may occur. A FWH device is not allowed to assert an Error SYNC. 	

There are several error cases that can occur on the LPC interface. The PCH responds as defined in Section 4.2.1.9 of the *Low Pin Count Interface Specification*, Revision 1.1 to the stimuli described therein. There may be other peripheral failure conditions; however, these are not handled by the PCH.

24.7.6 SYNC Error Indication

The PCH responds as defined in Section 4.2.1.10 of the *Low Pin Count Interface Specification* Revision 1.1.

Upon recognizing the SYNC field indicating an error, the PCH treats this as a SERR# by reporting this into the Device 31 Error Reporting Logic.



24.7.7 LFRAME# Usage

The PCH follows the usage of LFRAME# as defined in the *Low Pin Count Interface Specification*, Revision 1.1.

The PCH performs an abort for the following cases (possible failure cases):

- The PCH starts a Memory or I/O cycle, but no device drives a valid SYNC after four consecutive clocks.
- The PCH starts a Memory or I/O and the peripheral drives an invalid SYNC pattern.
- A peripheral drives an invalid value.

24.7.8 I/O Cycles

For I/O cycles targeting registers specified in the PCH's decode ranges, the PCH performs I/O cycles as defined in the *Low Pin Count Interface Specification*, Revision 1.1. These are 8-bit transfers. If the processor attempts a 16-bit or 32-bit transfer, the PCH breaks the cycle up into multiple 8-bit transfers to consecutive I/O addresses.

Note: If the cycle is not claimed by any peripheral (and subsequently aborted), the PCH returns a value of all 1s (FFh) to the processor. This is to maintain compatibility with ISA I/O cycles where Pull-up resistors would keep the bus high if no device responds.

24.7.9 LPC Power Management

24.7.9.1 LPCPD# Protocol

Same timings as SUS_STAT#. Upon driving SUS_STAT# low, the PCH drives LFRAME# low and tri-states (or drives low) LAD[3:0].

Note: The *Low Pin Count Interface Specification*, Revision 1.1 defines the LPCPD# protocol where there is at least 30 μ s from LPCPD# assertion to LRST# assertion. This specification explicitly states that this protocol only applies to entry/exit of low power states which does not include asynchronous reset events. The PCH asserts both SUS_STAT# (connects to LPCPD#) and PLTRST# (connects to LRST#) at the same time during a global reset. This is not inconsistent with the LPCPD# protocol.

24.7.10 Configuration and PCH Implications

24.7.10.1 LPC I/F Decoders

To allow the I/O cycles and memory mapped cycles to go to the LPC interface, the PCH includes several decoders. During configuration, the PCH must be programmed with the same decode ranges as the peripheral. The decoders are programmed using the D 31:F0 configuration space.

Note: The PCH cannot accept PCI write cycles from PCI-to-PCI bridges or devices with similar characteristics (specifically those with a "Retry Read" feature which is enabled) to an LPC device if there is an outstanding LPC read cycle towards the same PCI device or bridge. These cycles are not part of normal system operation, but may be encountered as part of platform validation testing using custom test fixtures.

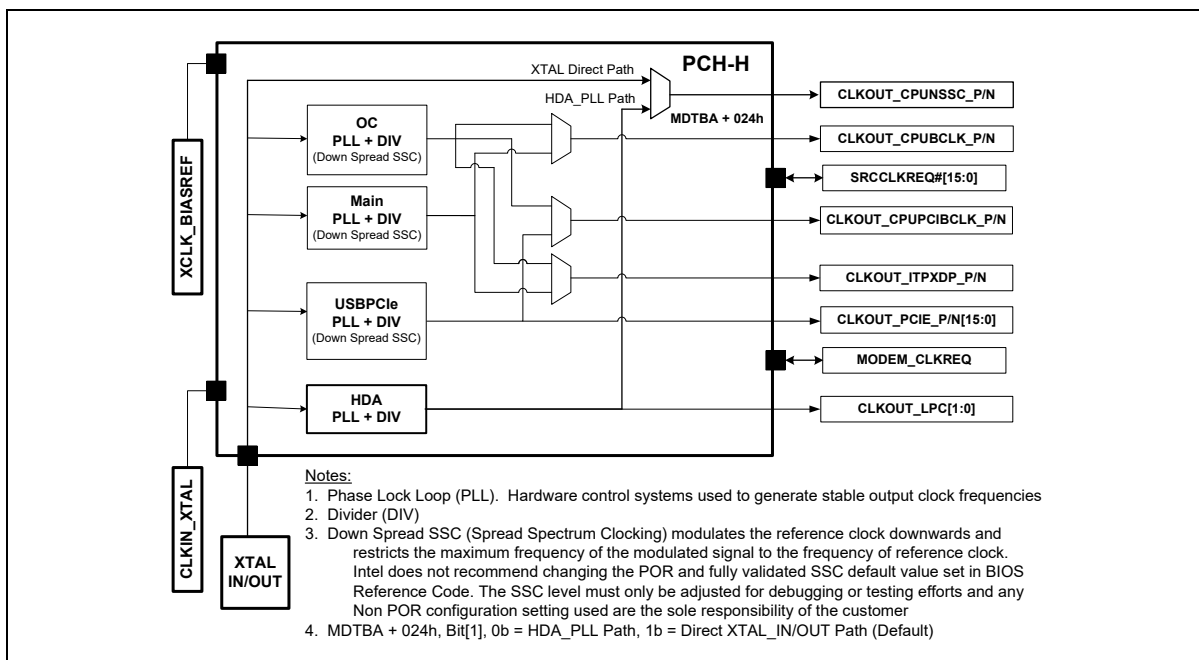


25 PCH and System Clocks

25.1 Overview

Platform Controller Hub (PCH) based platforms require several single-ended and differential clocks to synchronize signal operations and data propagations system wide between many interfaces and across multiple clock domains. The PCH generates and provides this complete system clocking solution through its Integrated Clock Controller (ICC).

Figure 25-1. Integrated Clock Controller (ICC) Diagram



25.2 PCH ICC Clocking Profiles

The PCH ICC hardware includes the following clocking profiles:

- “Standard” Profile (Figure 25-1)
 - OC PLL = Disabled
 - Main PLL = Enabled with Down Spread Spectrum Clcking (SSC)
 - USBPCIe PLL = Enabled with Down Spread Spectrum Clcking (SSC)
 - HDA PLL = Enabled
- “Adaptive” Profile (Figure 25-1)
 - OC PLL = Enabled with Down Spread Spectrum Clcking (SSC) and Under Clcking Capability
 - Main PLL = Disabled
 - USBPCIe PLL = Enabled with Down Spread Spectrum Clcking (SSC)
 - HDA PLL = Enabled

- “Over Clocking” Profile (Figure 25-1)
 - OC PLL = Enabled with Down Spread Spectrum Clocking (SSC) and Over Clocking Capability
 - Main PLL = Disabled
 - USBPCIe PLL = Enabled with Down Spread Spectrum Clocking (SSC)
 - HDA PLL = Enabled

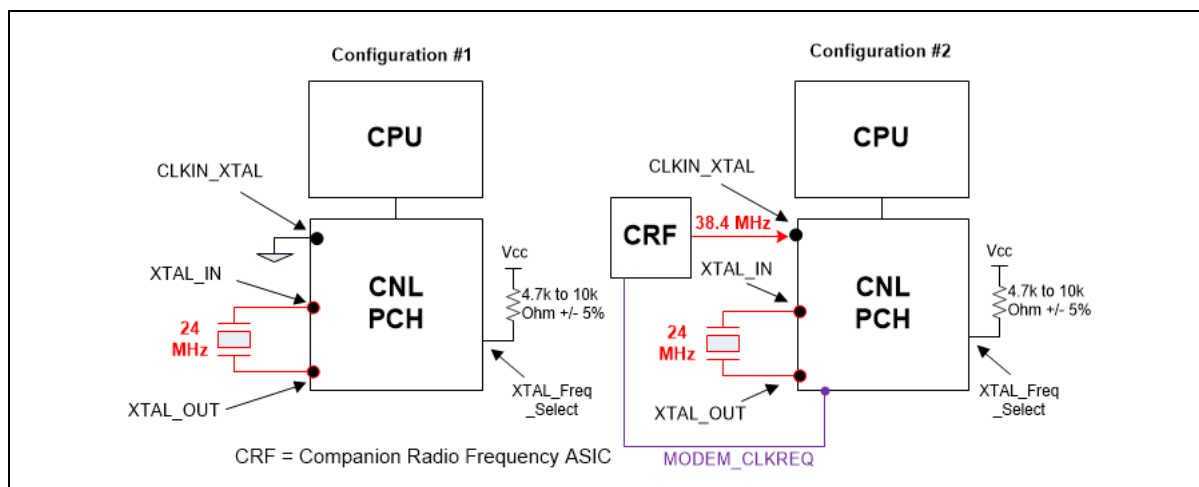
Note: The Standard ICC Profile is set by default and is the recommended ICC Clocking Profile.

Table 25-1. PCH ICC Clocking Profile Support

PCH SKU	Standard Profile Support	Adaptive Profile Support	Over Clocking Profile Support
H310	Yes	Yes	No
B360	Yes	Yes	No
H370	Yes	Yes	No
Q370	Yes	Yes	No
Z390	Yes	Yes	Yes
C242	Yes	Yes	No
C246	Yes	Yes	No
HM370	Yes	Yes	Yes
QM370	Yes	Yes	Yes
CM246	Yes	Yes	Yes

25.3 PCH ICC XTAL Input Configurations

Figure 25-2. PCH ICC XTAL Input Configurations



25.4 Signal Descriptions



Table 25-1. Signal Descriptions

Name	Type	SSC Capable	Description
CLKOUT_CPUNSSC_P CLKOUT_CPUNSSC_N	O	No	Filtered Crystal Reference Clock to CPU: 24 MHz differential, filtered crystal reference clock to the processor
CLKOUT_CPUBCLK_P CLKOUT_CPUBCLK_N	O	Yes	Differential Clock to CPU: Reference clock to the processor 100 MHz differential core
CLKOUT_CPUPCIBCLK_P CLKOUT_CPUPCIBCLK_N	O	Yes	Differential PCIe* Reference Clock to CPU: 100 MHz PCIe* 3.0 specification compliant differential PCIe* reference clock to the processor
CLKOUT_ITPXD_P CLKOUT_ITPXD_N	O	Yes	Differential ITP Debug Clock: 100 MHz differential output to processor XDP/ITP connector on the platform
CLKOUT_PCIE_P[15:0] CLKOUT_PCIE_N[15:0]	O	Yes	PCI Express* Clock Output: Serial Reference 100 MHz PCIe* 3.0 specification compliant differential output clocks to PCIe* devices
CLKOUT_LPC[1:0]	O	No	Low Pin Count (LPC) Clock Outputs: Single-Ended 24 MHz output to various single load connectors/devices
SRCCLKREQ#[15:0]	I/O		Clock Request: Serial Reference Clock request signals for PCIe* 100 MHz differential clocks
MODEM_CLKREQ	O		Aggregated XTAL CLKREQ: Used for Companion Radio Frequency ASIC (CRF) device initialization flow indication
XTAL_IN	I		Crystal Input: Input connection for 24 MHz crystal to PCH
XTAL_OUT	O		Crystal Output: Output connection for 24 MHz crystal to PCH
XCLK_BIASREF	I/O		Differential Clock Bias Reference: Used to set BIAS reference for differential clocks
CLKIN_XTAL	I		XTAL Clock Input: Single ended integrated CNV (Connectivity) XTAL clock input
XTAL_Freq_Select (GPP_J4)	I		XTAL Frequency Select: GPP_J4 Pin Strap for XTAL frequency selection. An external pull-up to VCC (1.8V) is required on this strap for 24 MHz XTAL operation
Notes:			
<ol style="list-style-type: none"> SSC = Spread Spectrum Clocking. Intel does not recommend changing the Plan of Record and fully validated SSC default value set in BIOS Reference Code. The SSC level must only be adjusted for debugging or testing efforts and any Non POR configuration setting used are the sole responsibility of the customer. The SRCCLKREQ#[15:0] signals can be configured to map to any of the PCH-H PCI Express* Root Ports SRCCLKREQ#[15:0] to CLKOUT_PCIE_P/N[15:0] Mapping Requirements <ul style="list-style-type: none"> SRCCLKREQ#[7:0] signals can be mapped to any of the CLKOUT_PCIE_P/N[7:0] differential clock pairs SRCCLKREQ#[15:8] signals can be mapped to any of the CLKOUT_PCIE_P/N[15:8] differential clock pairs 			

25.5 I/O Signal Pin States

Table 25-2. I/O Signal Pin States (Sheet 1 of 2)

Signal Name	S3/S4/S5	S0 Entry	S0	Deep Sx
CLKOUT_CPUNSSC_P CLKOUT_CPUNSSC_N	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)
CLKOUT_CPUBCLK_P CLKOUT_CPUBCLK_N	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)
CLKOUT_CPUPCIBCLK_P CLKOUT_CPUPCIBCLK_N	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)
CLKOUT_ITPXD_P CLKOUT_ITPXD_N	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)
CLKOUT_PCIE_P[15:0] CLKOUT_PCIE_N[15:0]	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)



Table 25-2. I/O Signal Pin States (Sheet 2 of 2)

Signal Name	S3/S4/S5	S0 Entry	S0	Deep Sx
CLKOUT_LPC[1:0]	OFF (Gated Low)	Bringing up the Clock	Toggling	OFF (Gated Low)
SRCCLKREQ#[15:0]	Un-driven	Un-driven	Driven	Off
MODEM_CLKREQ	Un-driven	Un-driven	Driven	Off

25.6 General Features

- The PCH Integrated Clock Controller (ICC) generates and supplies all the PCH reference clocks for internal needs and it provides the complete platform system clocking solution.
- All of the ICC PCH internal reference clocks and all of the single-ended and differential clock outputs are generated from an external 24 MHz crystal through the PCH XTAL_IN/OUT pins, where the crystal accuracy is required to be less than 100 ppm.
 - Note: ppm stands for parts per million, and it indicates how much a crystal's frequency may deviate from the nominal value.
- CLKOUT_PCIE_P/CLKOUT_PCIE_N 100MHz PCIe* 3.0 compliant differential output clocks support CLKREQ# based power management
- CLKOUT_LPC[1:0] single-ended output clocks support CLKRUN# based power management, they require no external loop back clock for internal logic, and they only support a single load configurations.
- System Power Management support includes shutdown of all PCH ICC Phase Locked Loops (PLL), PCH ICC internal and external clocks, and includes the shutdown of the external crystal source.

§ §



26 PCI Express* (PCIe*)

26.1 Overview

- PCH-H supports up to 16 PCIe* Ports (or devices) and 24 PCIe* Lanes, with transfer rates up to 8 GT/s (Gen3)
- Intel® Rapid Storage Technology (Intel® RST) for PCIe* Storage Devices
 - Remapped PCIe* NVMe SSD Non-Windows* Boot Operating System configurations are not 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)



26.2 Signal Description

PCH	Name	Type	Description
PCH-H	PCIE[24:1]_TXP PCIE[24:1]_TXN	O	PCI Express* Differential Transmit Pairs These are PCI Express* based outbound high-speed differential signals
	PCIE[24:1]_RXP PCIE[24:1]_RXN	I	PCI Express* Differential Receive Pairs These are PCI Express* based inbound high-speed differential signals
	PCIE_RCOMP PCIE_RCOMPN	I	Impedance Compensation Inputs

26.3 I/O Signal Planes and States

Table 26-1. Power Plane and States for PCI Express* Signals

Signal Name	Type	Power Plane	During Reset ²	Immediately After Reset ²	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_RCOMP PCIE_RCOMPN	I	Primary	Undriven	Undriven	Undriven	Off

Notes:

- PCIE_RXP\RXN pins transition from un-driven to Internal Pull-down during Reset.
- Reset reference for primary well pins is RSMRST#

26.4 PCI Express* Port Support Feature Details

Table 26-2. PCI Express* Port Feature Details

PCH	Max. Device (Ports)	Max. Lanes	PCIe* Gen Type	Encoding	Transfer Rate (MT/s)	Theoretical Max. Bandwidth (GB/s)		
						x1	x2	x4
PCH-H	16	24	1	8b/10b	2500	0.25	0.50	1.00
			2	8b/10b	5000	0.50	1.00	2.00
			3	128b/130b	8000	1.00	2.00	3.94

Notes:

- Theoretical Maximum Bandwidth (GB/s) = ((Transfer Rate * Encoding * # PCIe Lanes) / 8) / 1000
 — Gen3 Example: = ((8000 * 128/130 * 4) / 8) / 1000 = 3.94 GB/s
- When GbE is enabled on a PCIe Root Port, the Max. Device (Ports) value listed is reduced by a factor of 1



Figure 26-1. Supported PCI Express* Link Configurations

PCH-H Details		PCIe* Controller #1				PCIe* Controller #2				PCIe* Controller #3 Cycle Router #1				PCIe* Controller #4				PCIe* Controller #5 Cycle Router #3				PCIe* Controller #6 Cycle Router #2			
Flex I/O Lane #	PCIe* Lane #	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
HM370	1x4					RP 5				RP 9				RP 13				RP 17				RP 21			
	1x4 LR					RP 5				RP 9				RP 13				RP 17				RP 21			
	2x2					RP 5				RP 9				RP 13				RP 17				RP 21			
	1x2+2x1					RP 5				RP 9				RP 13				RP 17				RP 21			
	2x1+1x2					RP 5				RP 9				RP 13				RP 17				RP 21			
QM370	1x4					RP 5				RP 9				RP 13				RP 17				RP 21			
	1x4 LR					RP 5				RP 9				RP 13				RP 17				RP 21			
	2x2					RP 5				RP 9				RP 13				RP 17				RP 21			
	1x2+2x1					RP 5				RP 9				RP 13				RP 17				RP 21			
	2x1+1x2					RP 5				RP 9				RP 13				RP 17				RP 21			
CM246	1x4	RP 1				RP 5				RP 9				RP 13				RP 17				RP 21			
	1x4 LR	RP 1				RP 5				RP 9				RP 13				RP 17				RP 21			
	2x2	RP 1				RP 5				RP 9				RP 13				RP 17				RP 21			
	1x2+2x1	RP 1				RP 5				RP 9				RP 13				RP 17				RP 21			
	2x1+1x2	RP 1				RP 5				RP 9				RP 13				RP 17				RP 21			

PCH-H Details		PCIe* Controller #1				PCIe* Controller #2				PCIe* Controller #3 Cycle Router #1				PCIe* Controller #4				PCIe* Controller #5 Cycle Router #3				PCIe* Controller #6 Cycle Router #2			
Flex I/O Lane #	PCIe* Lane #	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
H310	1x4					RP 5																			
	1x4 LR					RP 5																			
	2x2					RP 5				RP 11															
	1x2+2x1					RP 5				RP 11															
	2x1+1x2					RP 5				RP 11															
B360	1x4					RP 5				RP 9								RP 21							
	1x4 LR					RP 5				RP 9								RP 21							
	2x2					RP 5				RP 9								RP 21							
	1x2+2x1					RP 5				RP 9								RP 21							
	2x1+1x2					RP 5				RP 9								RP 21							
H370 B365	1x4					RP 5				RP 9				RP 13				RP 21							
	1x4 LR					RP 5				RP 9				RP 13				RP 21							
	2x2	RP 3				RP 5				RP 9				RP 13				RP 21							
	1x2+2x1	RP 3				RP 5				RP 9				RP 13				RP 21							
	2x1+1x2	RP 3				RP 5				RP 9				RP 13				RP 21							
C246 Q370 Z390	1x4	RP 1				RP 5				RP 9				RP 13				RP 17							
	1x4 LR	RP 1				RP 5				RP 9				RP 13				RP 17							
	2x2	RP 1				RP 5				RP 9				RP 13				RP 17							
	1x2+2x1	RP 1				RP 5				RP 9				RP 13				RP 17							
	2x1+1x2	RP 1				RP 5				RP 9				RP 13				RP 17							
C242	1x4	RP 1								RP 9								RP 21							
	1x4 LR	RP 1								RP 9								RP 21							
	2x2	RP 1								RP 9								RP 21							
	1x2+2x1	RP 1								RP 9								RP 21							
	2x1+1x2	RP 1								RP 9								RP 21							

Notes:

1. RP# refers to a specific PCH PCI Express* Root Port #; for example RP3 = PCH PCI Express* Root Port 3
2. A PCIe* Lane is composed of a single pair of Transmit (TX) and Receive (RX) differential pairs, for a total of four data wires per PCIe* Lane (such as PCIE[3]_TXP/ PCIE[3]_TXN and PCIE[3]_RXP/ PCIE[3]_RXN make up PCIe Lane 3). A connection between two PCIe* devices is known as a PCIe* Link, and is built up from a collection of one or more PCIe* Lanes which make up the width of the link (such as bundling 2 PCIe* Lanes together would make a x2 PCIe* Link). A PCIe* Link is addressed by the lowest number PCIe* Lane it



- connects to and is known as the PCIe* Root Port (such as a x2 PCIe* Link connected to PCIe* Lanes 3 and 4 would be called x2 PCIe* Root Port 3).
3. The PCIe* Lanes can be configured independently from one another but the max number of configured Root Ports (Devices) must not be exceeded
 - PCH-H: A maximum of 16 PCIe* Root Ports (or devices) can be enabled
 - When a GbE Port is enabled, the maximum number of PCIe* Ports (or devices) that can be enabled reduces based off the following:
 - PCH-H: Max PCIe* Ports (or devices) = 16 - GbE (0 or 1)
 4. Unidentified lanes within a PCIe* Link Configuration are disabled but their physical lanes are used for the identified Root Port
 5. Supports up to Three Remapped (Intel® Rapid Storage Technology) PCIe* Storage Devices
 - Cells highlighted in Green identify controllers, configurations, and lanes that can be used for a x2/x4 Intel® Rapid Storage Technology Remapped PCIe* NVMe SSD or a x2/x4 Next Generation Intel® Optane™ Memory Device
 6. The SRCCLKREQ#[15:0] signals can be configured to map to any of the PCI Express* Root Ports
 7. SRCCLKREQ#[15:0] to CLKOUT_PCIE_P/N[15:0] Mapping Requirements
 - SRCCLKREQ#[7:0] signals can be mapped to any of the CLKOUT_PCIE_P/N[7:0] differential clock pairs
 - SRCCLKREQ#[15:8] signals can be mapped to any of the CLKOUT_PCIE_P/N[15:8] differential clock pairs
 8. Reference and understand the PCIe* High Speed I/O Multiplexing details covered in the "Flexible I/O" Chapter
 9. Lane Reversal Supported Motherboard PCIe* Configurations = 1x4 and 2x1+1x2
 - The 2x1+1x2 configuration is enabled by setting the PCIe* Controller soft straps to 1x2+2x1 with Lane Reversal Enabled
 - 1x4 = 1x4 with Lane Reversal Disabled, 1x4 LR = 1x4 with Lane Reversal Enabled
 10. For unused SATA/PCIe* Combo Lanes, Flex I/O Lanes that can be configured as PCIe* or SATA, the lanes must be statically assigned to SATA or PCIe* via the SATA/PCIe Combo Port Soft Straps . These unused SATA/PCIe* Combo Lanes must not be assigned as polarity based.
 - Refer to the "Flexible I/O" chapter for SATA/PCIe* Combo Lane identification

26.4.1 Intel® Rapid Storage Technology (Intel® RST) for PCIe* Storage

Intel® Rapid Storage Technology for PCIe* Storage includes the PCH PCIe* Controller Remapping Hardware, also referred to as Cycle Routers, and the Intel® RST Driver. The Remapping Hardware is a PCH PCIe* Controller architecture feature that works with the Intel® RST Driver to control and remap PCIe* storage devices to the PCH AHCI SATA Controller.

The PCH has multiple PCIe* Controllers where some, not all, of these Controllers have the Remapping Hardware. These specific PCIe* Controllers along with the Intel® RST Driver handle the remapping for x2 or x4 PCIe* storage devices. Special care must be taken to make sure the correct PCH PCIe* Lanes are used that are associated with these specific PCIe* Controllers. [Figure 26-2](#) describes and identifies all of the Intel® RST PCIe* controller, configuration, and lane support details.

26.4.2 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.

[Table 26-3](#) summarizes interrupt behavior for MSI and wire-modes. In the table "bits" refers to the hot-plug and PME interrupt bits.

**Table 26-3. 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

26.4.3 PCI Express* Power Management

26.4.3.1 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.

26.4.3.2 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. Refer [Section 26.4.3.3](#) for SMI/SCI generation.

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.

26.4.3.3 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.

26.4.3.4 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).

26.4.4 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 μ s.

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.

26.4.5 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.

26.4.6 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.

26.4.7 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

26.4.8 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.

26.4.8.1 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.

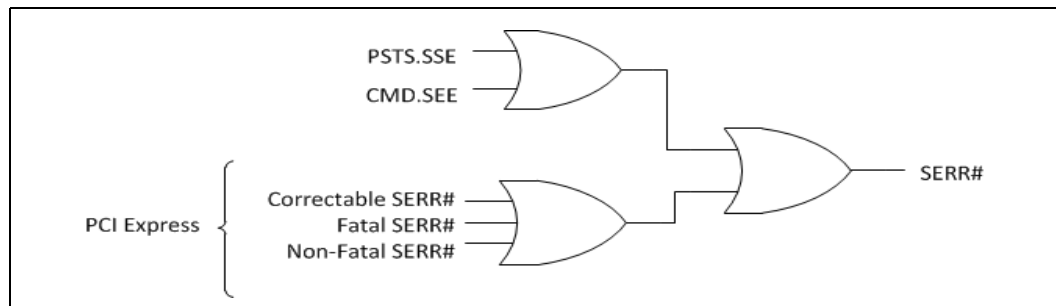
26.4.8.2 Access Control Services (ACS)

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.

26.4.9 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 26-2. Generation of SERR# to Platform



26.4.10 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



26.4.10.1 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.

26.4.10.2 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.

26.4.11 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”.

26.4.12 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 and 2x1+1x2
- 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

26.4.13 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.

Note: PCIe Root Ports transmit the lower byte [7:0] of the Propagation Delay Field first instead of the upper byte [31:24] within their PTM DelayResponseD (Response with Data) messages.

§ §



27 Power Management

27.1 Acronyms

Acronyms	Description
PMC	Power Management Controller
STD	Suspend To Disk
STR	Suspend To RAM
PMIC	Power Management Integrated Circuit
VR	Voltage Regulator

27.2 References

Specification	Location
Advanced Configuration and Power Interface (ACPI)	http://www.acpi.info/spec.htm

27.3 Overview

The Power Management Controller (PMC) is the PCH unit that handles all PCH power management related activities. This unit administers power management functions of the PCH including interfacing with other logic and controllers on the platform to perform power state transitions (such as SLP_S3# and PLTRST#); configure, manage and respond to wake events; aggregate and report latency tolerance information for devices and peripherals connected to and integrated into the PCH.

Note: DeepS3 is not supported in this PCH. In this document DeepSx refers to DeepS4 and DeepS5 states.

27.4 Signal Description

Name	Type	Description
ACPRESENT /GPD1	I	ACPRESENT: This input pin indicates when the platform is plugged into AC power or not. In addition to the previous Intel® CSME to EC communication, the PCH uses this information to implement the Deep Sx policies. For example, the platform may be configured to enter Deep Sx when in S4 or S5 and only when running on battery. This is powered by Deep Sx Well.
BATLOW# /GPD0	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 and exit from Deep Sx state. This signal can also be enabled to cause an SMI# when asserted. For Mobile package, this signal is multiplexed with GPD_0. This signal must be tied high to the VCCDSW_3P3, which will be tied to VCCPRIM_3P3 on Deep Sx disabled platforms. Notes: 1. Require external pull-up to VCCDSW_3P3. 2. The signal is intended for use on mobile platforms.



Name	Type	Description
CPU_C10_GATE# / GPP_J1	O	External Power Gate control for VCCIO, VCCSTG and VCCPLL_OC during C10. When asserted, VCCIO and VCCSTG can be 0V, however the power good indicators for these rails must remain asserted.
DRAM_RESET#	OD	System Memory DRAM Reset: Active low reset signal to DRAM. Note: An external pull-up to the DRAM power plane is required.
DSW_PWROK	I	DSW PWROK: Power OK Indication for the VCCDSW_3P3 voltage rail. This input is tied together with RSMRST# on platforms that do not support Deep Sx. Note: This signal is in the RTC well.
LAN_WAKE#/GPD2	I	LAN WAKE: is an active low wake indicator from the GbE PHY. Note: External pull-up required.
LANPHYPC /GPD11	O	LAN PHY Power Control: LANPHYPC is used to indicate that power needs to be restored to the Platform LAN Connect Device.
PCH_PWROK	I	PCH Power OK: When asserted, PCH_PWROK is an indication to the PCH that all of its core power rails have been stable for at least 5 ms. PCH_PWROK can be driven asynchronously. When PCH_PWROK is negated, the PCH asserts PLTRST#. Note: PCH_PWROK must not glitch, even if RSMRST# is low.
PLTRST#/GPP_B13	O	Platform Reset: The PCH asserts PLTRST# to reset devices on the platform (such as SIO, LAN, processor, and so forth.). The PCH asserts PLTRST# low in Sx states and when a cold, warm, or global reset occurs. The PCH de-asserts PLTRST# upon exit from Sx states and the aforementioned resets. There is no guaranteed minimum assertion time for PLTRST#. Note: PCIe* specification requires that the power rails associated with PCIe* (typically the 3.3V, 5V, and 12V core well rails) have been valid for 100 ms prior to PLTRST# de-assertion. System designers must ensure the requirement is met on the platform.
PME#/GPP_A11/SD_VDD2_PWR_EN#	I/OD	Power Management Event: Driven by devices to wake the system or issue SCI.
PS_ONB#/GPP_F_14	O	Platform PSU Control: Used to indicate to the PSU when to turn off its main rails to meet new California Energy Commission.
PWRBTN#/GPD3	I	Power Button: 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 PWRBTN# is pressed for more than 4 seconds (default; timing is configurable), this will cause an unconditional transition (power button override) to the S5 state. Override will occur even if the system is in the S3-S4 states. This signal has an internal Pull-up resistor and has an internal 16 ms de-bounce on the input. Note: Upon entry to S5 due to a power button override, if Deep Sx is enabled and conditions are met, the system will transition to Deep S5.
RSMRST#	I	Primary Well: This signal is used for resetting the resume power plane logic. This signal must be asserted for at least 10ms after the suspend power wells are valid. When de-asserted, this signal is an indication that the primary power wells are stable.
SLP_A#/GPD6	O	SLP_A#: Signal asserted when the Intel® CSME platform goes to M-Off or M3-PG. Depending on the platform, this pin may be used to control power to various devices that are part of the Intel® CSME sub-system in the platform.
SLP_LAN#	O	LAN Sub-System Sleep Control: When SLP_LAN# is de-asserted it indicates that the PHY device must be powered. When SLP_LAN# is asserted, power can be shut off to the PHY device. SLP_LAN# will always be de-asserted in S0 and anytime SLP_A# is de-asserted.
SLP_WLAN# / GPD9	O	WLAN Sub-System Sleep Control: When SLP_WLAN# is asserted, power can be shut off to the external wireless LAN device. SLP_WLAN# will always be de-asserted in S0. SLP_WLAN# shall not be used to shut down power to integrated connectivity (CNVi).



Name	Type	Description
SLP_S0#/GPP_B12	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 EC for other power management related optimizations.
SLP_S3#/GPD4	O	S3 Sleep Control: SLP_S3# 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.
SLP_S4#/GPD5	O	S4 Sleep Control: SLP_S4# 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. Note: This pin must be used to control the DRAM power in order to use the PCH DRAM power-cycling feature.
SLP_S5#/GPD10	O	S5 Sleep Control: SLP_S5# is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) states.
SLP_SUS#	O	Deep Sx Indication: When asserted (driven low), this signal indicates PCH is in Deep Sx state where primary power is shut off for enhanced power saving. When de-asserted (driven high), this signal indicates exit from Deep Sx state and primary power can be applied to PCH. If Deep Sx is not supported, then this pin can be left unconnected. Note: This pin is in the DSW power well.
SUSACK#/GPP_A15	I	SUSACK#: If Deep Sx is supported, the EC/motherboard controlling logic must change SUSACK# to match SUSWARN# once the EC/motherboard controlling logic has completed the preparations discussed in the description for the SUSWARN# pin. Note: SUSACK# is only required to change in response to SUSWARN# if Deep Sx is supported by the platform.
SUSCLK/GPD8	O	Suspend Clock: This clock is a digitally buffer version of the RTC clock.
SUSWARN#/SUSPWRDNACK/GPP_A13	O	SUSWARN#: This pin asserts low when the PCH is planning to enter the Deep Sx power state and remove Primary power (using SLP_SUS#). The EC/motherboard controlling logic must observe edges on this pin, preparing for primary well power loss on a falling edge and preparing for Primary well related activity (host/Intel® CSME wakes and runtime events) on a rising edge. SUSACK# must be driven to match SUSWARN# once the above preparation is complete. SUSACK# should be asserted within a minimal amount of time from SUSWARN# assertion as no wake events are supported if SUSWARN# is asserted but SUSACK# is not asserted. Platforms supporting Deep Sx, but not wishing to participate in the handshake during wake and Deep Sx entry may tie SUSACK# to SUSWARN#. This pin is multiplexed with SUSPWRDNACK since it is not needed in Deep Sx supported platforms.
SUSPWRDNACK/SUSWARN#/GPP_A13	O	SUSPWRDNACK: Active high. Asserted by the PCH on behalf of the Intel® CSME when it does not require the PCH Primary well to be powered. Platforms are not expected to use this signal when the PCH Deep Sx feature is used.
SX_EXIT_HOLDOFF#/GPP_A12 /ISH_GP6 / BM_BUSY#	I	Sx Exit Holdoff Delay: Delay exit from Sx state after SLP_A# is de-asserted. Refer Section 27.7.8.5 for more details. Note: When eSPI is enabled, SX_EXIT_HOLDOFF# functionality is not available, and assertion of the signal will not impact Sx exit flows.
SYS_PWROK	I	System Power OK: This generic power good input to the PCH is driven and utilized in a platform-specific manner. While PCH_PWROK always indicates that the primary wells of the PCH are stable, SYS_PWROK is used to inform the PCH that power is stable to some other system component(s) and the system is ready to start the exit from reset.
SYS_RESET#	I	System Reset: This pin forces an internal reset after being de-bounced.
VRALERT#/GPP_B2	I	VR Alert: ICC Max. throttling indicator from the PCH voltage regulators. VRAlert# pin allows the VR to force throttling to prevent an over current shutdown.



Name	Type	Description
WAKE#	I/OD	PCI Express* Wake Event in Sx: Input Pin in Sx. Sideband wake signal on PCI Express* asserted by components requesting wake up. Note: This is Output pin during S0ix states hence this pin can not be used to wake up the system during S0ix states. Note: External pull-up required.
CLKRUN#/GPP_A8	I/OD	LPC Clock Run: Used to control CLKOUT_LPC[1:0]. Connects to peripherals that need to request clock restart or prevention of clock stopping.
SUS_STAT#/ ESPI_RESET#/GPP_A14	O	LPC Mode - Suspend Status: This signal is asserted by the PCH to indicate that the system will be entering a low power state soon. This can be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It can also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes. Note: In eSPI Mode, this signal functions as ESPI Reset#. Reset signal from PCH to eSPI slave.

27.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
ACPRESENT	Pull-down	15 kohm - 40 kohm	1
LAN_WAKE#	Pull-down	15 kohm - 40 kohm	1
PWRBTN#	Pull-up	20 kohm +/- 30%	
SUSACK#	Pull-up	20 kohm +/- 30%	
WAKE#	Pull-down	15 kohm - 40 kohm	1
Notes: 1. Pull-down is configurable and can be enabled in Deep Sx state; refer to DSX_CFG register for more details.			

27.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ¹⁸	Immediately after Reset ¹⁸	S3/S4/S5	Deep Sx
ACPRESENT^{6,10,15}	DSW	Undriven / Driven Low ⁴	Undriven	Undriven	Undriven/ Internal Pull-down ⁸
BATLOW#	DSW	Undriven	Undriven	Undriven	OFF
DRAM_RESET#	DSW	Undriven	Undriven	Undriven	Undriven
DSW_PWROK	RTC	Undriven	Undriven	Undriven	Undriven
LANPHYPC^{10,16}	DSW	Driven Low	Driven Low	Driven Low	Driven Low
LAN_WAKE#¹⁵	DSW	Undriven	Undriven	Undriven	Undriven/ Internal Pull-down ⁸
PCH_PWROK	RTC	Undriven	Undriven	Undriven	Undriven
PLTRST#¹⁶	Primary	Driven Low	Driven High	Driven Low	OFF
PME#¹⁵	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	OFF
PWRBTN#¹⁵	DSW	Internal Pull-up	Internal Pull-up	Internal Pull-up	Internal Pull-up



Signal Name	Power Plane	During Reset ¹⁸	Immediately after Reset ¹⁸	S3/S4/S5	Deep Sx
RSMRST#	RTC	Undriven	Undriven	Undriven	Undriven
SUSACK# ¹⁵	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	OFF
SUSCLK ^{10,16}	DSW	Driven Low	Toggling	Toggling	Toggling ¹⁰
SUSWARN#/ SUSPWRDNACK ^{6,10, 16}	Primary	Driven Low	Driven Low	Driven Low ⁵	OFF
SLP_A# ^{6,16}	DSW	Driven Low	Driven High	Driven High/ Driven Low ¹²	Driven High/ Driven Low ¹²
SLP_S0# ^{1,17}	Primary	Driven High	Driven High	Driven High	OFF
SLP_S3# ^{6,16}	DSW	Driven Low	Driven High	Driven Low	Driven Low
SLP_S4# ^{6,16}	DSW	Driven Low	Driven High	Driven High/ Driven Low ²	Driven High/ Driven Low ⁹
SLP_S5# ^{6,16}	DSW	Driven Low	Driven High	Driven High/ Driven Low ³	Driven High/ Driven Low ⁹
SLP_LAN# ⁶	DSW	Driven Low	Driven Low	Driven High/ Driven Low ⁷	Driven High/ Driven Low ⁷
SLP_WLAN# ^{6,16}	DSW	Driven Low	Driven Low	Driven High/ Driven Low ⁷	Driven High/ Driven Low ⁷
SLP_SUS# ⁶	DSW	Driven Low	Driven High	Driven High	Driven Low
SX_EXIT_HOLDOFF# ¹⁵	Primary	Undriven	Undriven	Undriven	OFF
SYS_PWROK ¹³	Primary	Undriven	Undriven	Undriven	OFF
SYS_RESET# ¹³	Primary	Undriven	Undriven	Undriven	OFF
VRALERT# ¹⁵	Primary	Undriven	Undriven	Undriven	OFF
WAKE# ¹³	DSW	Undriven	Undriven	Undriven	Undriven/ Internal Pull-down

Notes:

1. Driven High during S0 and driven Low during S0 CS.
2. SLP_S4# is driven high in S3, driven low in S4/S5.
3. SLP_S5# is driven high in S3/S4, driven low in S5.
4. In non-Deep Sx mode, pin is driven low.
5. Based on wake events and Intel® CSME state. SUSPWRDNACK is always '0' while in M0 or M3, but can be driven to '0' or '1' while in M0ff state. SUSPWRDNACK is the default mode of operation. If Deep Sx is supported, then subsequent boots will default to SUSWARN#.
6. The pin requires glitch-free output sequence. The pad should only be pulled low momentarily when the corresponding buffer power supply is not stable.
7. Based on wake event and Intel® CSME state.
8. Pull-down is configurable and can be enabled in Deep Sx state; refer to DSX_CFG register for more details.
9. When platform enters Deep Sx, the SLP_S4# and SLP_S5# pin will retain the value it held prior to Deep Sx entry.
10. Internal weak pull-down resistor is enabled during power sequencing, but configurable (pull-down /pull-up/none) after boot.
11. N/A
12. Pin state is a function of whether the platform is configured to have Intel® CSME on or off in Sx.
13. Output High-Z, not glitch free.
14. Output High-Z, glitch free with ~120 kohm Pull-down during respective power sequencing
15. Output High-Z, not glitch free.
16. Output High-Z, glitch free with ~20 kohm Pull-down during respective power sequencing.
17. Output High-Z, glitch free with ~20 kohm Pull-up during respective power sequencing.
18. Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.



27.7 Functional Description

27.7.1 Features

- Support for *Advanced Configuration and Power Interface (ACPI)* providing power and thermal management
 - ACPI 24-Bit Timer SCI and SMI# Generation
- PCI PME# signal for Wake Up from Low-Power states
- System Sleep State Control
 - ACPI S3 state – Suspend to RAM (STR)
 - ACPI S4 state – Suspend-to-Disk (STD)
 - ACPI G2/S5 state – Soft Off (SOFF)
 - Power Failure Detection and Recovery
 - Deep Sx
- Intel® Management Engine Power Management Support
 - Wake events from the Intel® Management Engine (enabled from all S-States including Catastrophic S5 conditions)
- SLP_S0# signal for external platform VR power gating or EC power management handling during lower power condition

Note: Power Management timers accuracy is ~4.2% which will affect energy reporting counters, power button override duration etc.

27.7.2 PCH and System Power States

Below table shows the power states defined for PCH-based platforms. The state names generally match the corresponding ACPI states.

Table 27-1. General Power States for Systems Using the PCH (Sheet 1 of 2)

State / Substates	Legacy Name/Description
G0/S0/C0	Full On: Processor operating. Individual devices may be shut down or be placed into lower power states to save power.
G0/S0/Cx	Cx State: Cx states are processor power states within the S0 system state that provide for various levels of power savings. The processor manages c-state itself. The actual c-state is not passed to the PCH. Only c-state related messages are sent to the PCH and PCH will base its behavior on the actual data passed.
G1/S3	Suspend-To-RAM (STR): The system context is maintained in system DRAM, but power is shut off to non-critical circuits. Memory is retained and refreshes continue. All external clocks stop except RTC.
G1/S4	Suspend-To-Disk (STD): The context of the system is maintained on the disk. All power is then shut off to the system except for the logic required to resume.
G2/S5	Soft Off (SOFF): System context is not maintained. All power is shut off except for the logic required to restart. A full boot is required when waking.
S0ix	S0 idle states are often referred as S0ix states. CPU PKG C-states and platform latency tolerance will decide when to take the aggressive power management actions.
Deep Sx	Deep Sx: An optional low power state where system context may or may not be maintained depending upon entry condition. All power is shut off except for minimal logic that allows exiting Deep Sx. If Deep Sx state was entered from S4 state, then the resume path will place system back into S4. If Deep Sx state was entered from S5 state, then the resume path will place system back into S5.



Table 27-1. General Power States for Systems Using the PCH (Sheet 2 of 2)

State / Substates	Legacy Name/Description
G3	Mechanical OFF (M-Off): System context not maintained. All power is shut off except for the RTC. No "Wake" events are possible. This state occurs if the user removes the main system batteries in a mobile system, turns off a mechanical switch, or if the system power supply is at a level that is insufficient to power the "waking" logic. When system power returns, transition will depend on the state just prior to the entry to G3 and the AFTERG3_EN bit in the General Power Management Configuration (GEN_PMC0N) register. Refer to Table 27-7 for more details.

[Table 27-2](#) shows the transitions rules among the various states.

Note: Transitions among the various states may appear to temporarily transition through intermediate states. For example, in going from S0 to S4, it may appear to pass through the G1/S3 state. These intermediate transitions and states are not listed in the [Table 27-2](#).

Table 27-2. State Transition Rules for the PCH

Present State	Transition Trigger	Next State
G0/S0/C0	<ul style="list-style-type: none"> • DMI Msg • SLP_EN bit set • Power Button Override^{3,5} • Mechanical Off/Power Failure 	<ul style="list-style-type: none"> • G0/S0/Cx • G1/Sx or G2/S5 state • G2/S5 • G3
G0/S0/Cx	<ul style="list-style-type: none"> • DMI Msg • Power Button Override^{3,5} • Mechanical Off/Power Failure 	<ul style="list-style-type: none"> • G0/S0/C0 • S5 • G3
G1/S3	<ul style="list-style-type: none"> • Any Enabled Wake Event • Power Button Override^{3,5} • Mechanical Off/Power Failure 	<ul style="list-style-type: none"> • G0/S0/C0² • G2/S5 • G3
G1/S4	<ul style="list-style-type: none"> • Any Enabled Wake Event • Power Button Override^{3,5} • Conditions met as described in Section 27.7.7.6.1 and Section 27.7.7.6.2 • Mechanical Off/Power Failure 	<ul style="list-style-type: none"> • G0/S0/C0² • G2/S5 • Deep S4 • G3
G2/S5	<ul style="list-style-type: none"> • Any Enabled Wake Event • Conditions met as described in Section 27.7.7.6.1 and Section 27.7.7.6.2 • Mechanical Off/Power Failure 	<ul style="list-style-type: none"> • G0/S0/C0² • Deep S5 • G3
G2/Deep Sx	<ul style="list-style-type: none"> • Any Enabled Wake Event • ACPRESENT Assertion • Mechanical Off/Power Failure • Power Button Override 	<ul style="list-style-type: none"> • G0/S0/C0² • G1/S4 or G2/S5 (Refer Section 27.7.7.6.2) • G3 • G2/S5
G3	<ul style="list-style-type: none"> • Power Returns 	<ul style="list-style-type: none"> • S0/C0 (reboot) or G2/S5⁴ (stay off until power button pressed or other wake event)^{1,2}
<p>Notes:</p> <ol style="list-style-type: none"> 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 BATLOW# 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. 5. Upon entry to S5 due to a power button override, if Deep S5 is enabled and conditions are met per Section 27.7.7.6, the system will transition to Deep S5. 		

27.7.3 System Power Planes

The system has several independent power planes, as described in Table 27-3.

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

Table 27-3. System Power Plane

Plane	Controlled By	Description
Processor	SLP_S3# signal	The SLP_S3# signal can be used to cut the power to the processor completely.
Main (Applicable to Platform, PCH does not have a Main well)	SLP_S3# signal	When SLP_S3# 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, LPC I/F, and 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 SLP_S3# signal, but instead by the SLP_SUS# signal.
Memory	SLP_S4# signal SLP_S5# signal	When SLP_S4# goes active, power can be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory can also be shut down. When SLP_S5# goes active, power can be shut off to any circuit not required to wake the system from the S5 state. Since the memory context does not need to be preserved in the S5 state, the power to the memory can also be shut.
Intel® CSME	SLP_A#	SLP_A# signal is asserted when the Intel® CSME platform goes to M-Off or M3-PG. Depending on the platform, this pin may be used to control power to various devices that are part of the Intel® CSME sub-system in the platform.
LAN	SLP_LAN#	This signal is asserted in Sx/M-Off or Sx/M3-PG when both host and Intel® CSME WoL are not supported. This signal can be used to control power to the Intel GbE PHY.
Primary/ Suspend Well	SLP_SUS#	This signal is asserted when the Primary/Suspend rails can be externally shut off for enhanced power saving.
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.

27.7.4 SMI# /SCI Generation

Upon any enabled SMI event taking place while the End of SMI (EOS) bit is set, the PCH will clear the EOS bit and assert SMI to the processor, which will cause it to enter SMM space. SMI assertion is performed using a Virtual Legacy Wire (VLW) message. Prior system generations (those based upon legacy processors) used an actual SMI# pin.

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

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



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

Table 27-4 shows which events can cause an SMI and SCI.

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

Table 27-4. Causes of SMI and SCI (Sheet 1 of 2)

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



Table 27-4. Causes of SMI and SCI (Sheet 2 of 2)

Cause	SCI	SMI	Additional Enables (Note 1)	Where Reported
SMBus Host Controller	No	Yes	SMB_SMI_EN, Host Controller Enabled	SMBus host status reg.
SMBus Slave SMI message	No	Yes	None	SMBUS_SMI_STS
SMBus SMBALERT# signal active	No	Yes	None	SMBUS_SMI_STS
SMBus Host Notify message received	No	Yes	HOST_NOTIFY_INTREN	SMBUS_SMI_STS, HOST_NOTIFY_STS
BATLOW# assertion	Yes	Yes	BATLOW_EN=1	BATLOW_STS
Access microcontroller 62h/66h	No	Yes	MCSMI_EN	MCSMI_STS
SLP_EN bit written to 1	No	Yes	SMI_ON_SLP_EN=1	SMI_ON_SLP_EN_STS
SPI Command Completed	No	Yes	None	SPI_SMI_STS
eSPI SCI/SMI Request	Yes	Yes	eSPI_SCI_EN For SMI, refer eSPI section	eSPI_SCI_STS eSPI_SMI_STS
Software Generated GPE	Yes	Yes	SWGPE_EN=1	SWGPE_STS
Intel® CSME	Yes	Yes	ME_SCI_EN=1 ME_SCI_EN=0; ME_SMI_EN=1;	ME_SCI_STS ME_SMI_STS
GPIO Lockdown Enable bit changes from '1' to '0'	No	Yes	GPIO_UNLOCK_SMI_EN=1	GPIO_UNLOCK_SMI_STS
USB 3.1(xHCI) SMI Event	No	Yes	XHCI_SMI_EN=1	XHCI_SMI_STS
Wake Alarm Device Timer	Yes	Yes	WADT_EN	WADT_STS
ISH	Yes	No	ISH_EN	ISH_STS
RTC update-in-progress	No	Yes	Refer I/O Trap Register section	RTC_UIP_SMI_STS
SIO SMI events	No	Yes	SIP_SMI_EN	SIO_SMI_STS
SCC	No	Yes	SCC_SMI_EN	SCC_SMI_STS
Notes:				
<ol style="list-style-type: none"> 1. SCI_EN must be 1 to enable SCI, except for BIOS_RLS. SCI_EN must be 0 to enable SMI. 2. SCI can be routed to cause interrupt 9:11 or 20:23 (20:23 only available in APIC mode). 3. GBL_SMI_EN must be 1 to enable SMI. 4. EOS must be written to 1 to re-enable SMI for the next 1. 5. The PCH must have SMI fully enabled when the PCH is also enabled to trap cycles. If SMI is not enabled in conjunction with the trap enabling, then hardware behavior is undefined. 6. When a power button override first occurs, the system will transition immediately to S5. The SCI will only occur after the next wake to S0 if the residual status bit (PRBTNOR_STS) is not cleared prior to setting SCI_EN. 7. GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place. 8. Refer to GPIO chapter for specific GPIOs enabled for SCIs and/or SMIs 				

27.7.4.1 PCI Express* SCI

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

27.7.4.2 PCI Express* Hot-Plug

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



27.7.5 C-States

PCH-based systems implement C-states by having the processor control the states. The chipset exchanges messages with the processor as part of the C-state flow, but the chipset does not directly control any of the processor impacts of C-states, such as voltage levels or processor clocking.

27.7.6 Dynamic 24 MHz Clock Control

The 24 MHz clock can be dynamically controlled independent of any other low-power state.

The Dynamic 24 MHz Clock control is handled using the following signal:

CLKRUN#: Used by LPC peripherals or other legacy devices to request the system 24 MHz clock to run.

27.7.6.1 Conditions for Checking the 24 MHz Clock

When there is a lack of activity, the PCH has the capability to stop the 24 MHz clocks to conserve power. "Clock activity" is defined as any activity that would require the 24 MHz clock to be running.

Any of the following conditions will indicate that it is **not okay** to stop the 24 MHz clock:

- Cycles on LPC
- SERIRQ activity

27.7.6.2 Conditions for Maintaining the 24 MHz Clock

LPC or any other devices that wish to maintain the 24 MHz clock running will observe the CLKRUN# signal de-asserted, and then must re-assert if (drive it low) within 92 clocks.

- When the PCH has tri-stated the CLKRUN# signal after de-asserting it, the PCH then checks to view if the signal has been re-asserted (externally).
- After observing the CLKRUN# signal asserted for 1 clock, the PCH again starts asserting the signal.
- If an internal device needs the PCI bus, the PCH asserts the CLKRUN# signal.

27.7.6.3 Conditions for Stopping the 24 MHz Clock

- When there is a lack of activity (as defined above) for 29 to 30 clock cycles, the PCH de-asserts (drive high) CLKRUN# for 1 clock cycle and then tri-states the signal.
- If no device drives CLKRUN# low within 6 clock cycles after it has been de-asserted, the PCH will stop the 24 MHz clocks.

27.7.6.4 Conditions for Re-starting the 24 MHz Clock

- A peripheral asserts CLKRUN# to indicate that it needs the 24 MHz clock re-started.
- Observing the CLKRUN# signal asserted externally for 1 (free running) clock, the PCH again starts driving CLKRUN# asserted.



If an internal source requests the clock to be re-started, the PCH re-asserts CLKRUN#, then the PCH will start the 24 MHz clocks.

27.7.7 Sleep States

27.7.7.1 Sleep State Overview

The PCH directly supports different sleep states (S3–S5), which are entered by methods such as setting the SLP_EN bit or due to a Power Button press. The entry to the Sleep states is based on several assumptions:

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

27.7.7.2 Initiating Sleep State

Sleep states (S3–S5) are initiated by:

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

Table 27-5. Sleep Types

Sleep Type	Comment
S3	The PCH asserts SLP_S3#. The SLP_S3# signal controls the power to non-critical circuits. Power is only retained to devices needed to wake from this sleeping state, as well as to the memory.
S4	The PCH asserts SLP_S3# and SLP_S4#. The SLP_S4# signal shuts off the power to the memory subsystem. Only devices needed to wake from this state should be powered.
S5	The PCH asserts SLP_S3#, SLP_S4# and SLP_S5#.

27.7.7.3 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 27-6](#).

Note: 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 27-6. Causes of Wake Events

Cause	How Enabled	Wake from Sx	Wake from Deep Sx	Wake from Sx After Power Loss (Note 2)	Wake from "Reset" Types (Note 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
Any GPIOs except DSW GPIOs can be enabled for wake ⁵		Yes	No	No	No
LAN_WAKE#	Enabled natively (unless pin is configured to be in GPIO mode)	Yes	Yes	Yes	Yes
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#	(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
Intel® CSME Non-Maskable Wake	Always enabled as a wake event.	Yes	No	Yes	Yes
Integrated WoL Enable Override	WoL Enable Override bit (in Configuration Space).	Yes	Yes	Yes	Yes
Wake Alarm Device	WADT_EN in GPE0_EN[127:96]	Yes	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® CSME -initiated power button override, Intel® CSME -initiated host partition reset with power down, Intel® CSME Watchdog Timer, SMBus unconditional power down, processor thermal trip, 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.					



27.7.7.4 PCI Express* WAKE# Signal and PME Event Message

PCI Express* ports can wake the platform from any sleep state (S3, S4, or S5 or Deep Sx) using the WAKE# pin. WAKE# is treated as a wake event, but does not cause any bits to go active in the GPE_STS register.

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

27.7.7.5 Sx-G3-Sx, Handling Power Failures

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

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

1. **PWRBTN#:** PWRBTN# is always enabled as a wake event. When PCH_DPWROK is low (G3 state), the PWRBTN_STS bit is reset. When the PCH exits G3 after power returns (PCH_DPWROK goes high), the PWRBTN# signal will transition high due internal Pull-up, unless there is an on-board Pull-up/Pull-down) and the PWRBTN_STS bit is 0.
2. **RTC Alarm:** The RTC_EN bit is in the RTC well and is preserved after a power loss. Like PWRBTN_STS the RTC_STS bit is cleared when PCH_DPWROK goes low.
3. Any enabled wake event that was preserved through the power failure.

DSW_PWROK going low would place the PCH into a G3 state.

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

Table 27-7. Transitions Due to Power Failure

State at Power Failure	AFTERG3_EN Bit	Transition when Power Returns
S0, S3	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0
Deep S4	1 0	Deep S4 ¹ S0
Deep S5	1 0	Deep S5 ¹ S0

Note:

1. Entry state to Deep Sx is preserved through G3 allowing resume from Deep Sx to take appropriate path (that is, return to S4 or S5).
2. G3 related Power Failure is defined as DSW_PWROK transition low.



27.7.7.6 Deep Sx

To minimize power consumption while in S4/S5, the PCH supports a lower power, lower featured version of these power states known as Deep Sx. In the Deep Sx state, the Suspend wells are powered off, while the Deep Sx Well (DSW) remains powered. A limited set of wake events are supported by the logic located in the DSW.

The Deep Sx capability and the SUSWRDNACK pin functionality are mutually exclusive.

27.7.7.6.1 Entry Into Deep Sx

A combination of conditions is required for entry into Deep Sx.

All of the following must be met:

1. Intel® CSME in M-Off or M3-PG, AND
2. Either a. or b. as defined below
 - a. (S4AC_GATE_SUS AND S4) OR (S5AC_GATE_SUS AND S5)), OR
 - b. ((AC_PRESENT = 0) AND ((S3DC_GATE_SUS AND S3) OR (S4DC_GATE_SUS AND S4) OR (S5DC_GATE_SUS AND S5)))

Note: If REQ_CNV_NOWAKE_DSX is set to '1,' connectivity wake must be disabled in addition to the above DeepSx entry conditions to allow DeepSx entry.

Note: If REQ_BATLOW_DSX is set to '1,' BATLOW# must be asserted in addition to the above DeepSx entry conditions to allow DeepSx entry.

Table 27-8. Supported Deep Sx Policy Configurations

Configuration	S4DC_GATE_SUS	S4AC_GATE_SUS	S5DC_GATE_SUS	S5AC_GATE_SUS
1. Enabled in S5 when on Battery (ACPRESENT = 0)	0	0	1	0
2. Enabled in S5 (ACPRESENT not considered)	0	0	1	1
3. Enabled in S4 and S5 when on Battery (ACPRESENT = 0)	1	0	1	0
4. Enabled in S4 and S5 (ACPRESENT not considered)	1	1	1	1
5. Enabled in S4 and S5 when on Battery (ACPRESENT = 0)	1	0	1	0
6. Enabled in S4 and S5 (ACPRESENT not considered)	1	1	1	1
7. Deep S4/ S5 disabled	0	0	0	0
Note: All other configurations are RESERVED.				

The PCH also performs a SUSWARN#/SUSACK# handshake to ensure the platform is ready to enter Deep Sx. The PCH asserts SUSWARN# as notification that it is about to enter Deep Sx. Before the PCH proceeds and asserts SLP_SUS#, the PCH waits for SUSACK# to assert.

27.7.7.6.2 Exit from Deep Sx

While in Deep Sx, the PCH monitors and responds to a limited set of wake events (RTC Alarm, Power Button and WAKE#). Upon sensing an enabled Deep Sx wake event, the PCH brings up the Suspend well by de-asserting SLP_SUS#.

Table 27-9. Deep Sx Wake Events

Event	Enable
RTC Alarm	RTC_EN bit in PM1_EN_STS Register
Power Button	Always enabled
PCIe* WAKE# pin	PCIEXP_WAKE_DIS
Wake Alarm Device	WADT_EN in GPE0_EN
LAN_WAKE#	Enabled natively (unless the pin is configured to be in the GPIO mode)

ACPRESENT has some behaviors that are different from the other Deep Sx wake events. If the Intel® CSME has enabled ACPRESENT as a wake event then it behaves just like any other Intel® CSME Deep Sx wake event. However, even if ACPRESENT wakes are not enabled, if the Host policies indicate that Deep Sx is only supported when on battery, then ACPRESENT going high will cause the PCH to exit Deep Sx. In this case, the Suspend wells gets powered up and the platform remains in Sx/M-Off or Sx/M3-PG. If ACPRESENT subsequently drops (before any Host or Intel® CSME wake events are detected), the PCH will re-enter Deep Sx.

27.7.8 Event Input Signals and Their Usage

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

27.7.8.1 PWRBTN# (Power Button)

The PCH PWRBTN# signal operates as a “Fixed Power Button” as described in the *Advanced Configuration and Power Interface Specification*. PWRBTN# signal has a 16 ms de-bounce on the input. The state transition descriptions are included in [Table 27-10](#).

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

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



The PCH also supports modifying the length of time the Power Button must remain asserted before the unconditional power down occurs (4-14 seconds). The length of the Power Button override duration has no impact on the “extension” of the power button override timer while SLP_* stretching is in progress. The extended power button override period while stretching is in progress remains 9-10 seconds in all cases.

Table 27-10. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI or SCI generated (depending on SCI_EN, PWRBTN_EN and GLB_SMI_EN)	Software typically initiates a Sleep state Note: Processing of transitions starts within 100 us of the PWRBTN# input pin to PCH going low. ¹
S3 – S5	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup Note: Could be impacted by SLP_* min assertion. The minimum time the PWRBTN# pin should be asserted is 150 us. The PCH will start processing this change once the minimum time requirement is satisfied. ¹
Deep Sx	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup Note: Could be impacted by SLP_* min assertion. The minimum time the PWRBTN# pin should be asserted is 150 us. The PCH will start processing this change once the minimum time requirement is satisfied but subsequently the PWRBTN# pin needs to de-assert for at least 500 us after RSMRST# de-assertion otherwise the system waits indefinitely in S5 state. ¹
G3	PWRBTN# pressed	None	No effect since no power Not latched nor detected Note: During G3 exit, PWRBTN# pin must be kept de-asserted for a minimum time of 500 us after the RSMRST# has de-asserted. ² Note: Beyond this point, the minimum time the PWRBTN# pin has to be asserted to be registered by PCH as a valid wake event is 150 us. ¹
S0 – S4	PWRBTN# held low for at least 4 ³ consecutive seconds	Unconditional transition to S5 state and if Deep Sx is enabled and conditions are met per Section 27.7.7.6 , the system will then transition to Deep Sx.	No dependence on processor or any other subsystem Note: Due to internal PCH latency, it could take up to an additional ~1.3s after PWRBTN# has been held low for 4s before the system would begin transitioning to S5.
Notes:			
1. If PM_CFG.PB_DB_MODE='0', the debounce logic adds 16 ms to the start/minimum time for processing of power button assertions.			
2. This minimum time is independent of the PM_CFG.PB_DB_MODE value.			
3. The amount of time PWRBTN# must be asserted is configurable via PM_CFG2.PBOP. 4 seconds is the default.			

Power Button Override Function

If PWRBTN# is observed active for at least four consecutive seconds (always sampled after the output from debounce logic), the PCH should unconditionally transition to the G2/S5 state or Deep Sx, regardless of present state (S0 – S4), even if the PCH_PWROK



is not active. In this case, the transition to the G2/S5 state or Deep Sx does not depend on any particular response from the processor, nor any similar dependency from any other subsystem.

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

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

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

Sleep Button

The *Advanced Configuration and Power Interface Specification* defines an optional Sleep button. It differs from the power button in that it only is a request to go from S0 to S3–S4 (not S5). Also, in an S5 state, the Power Button can wake the system, but the Sleep Button cannot.

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

27.7.8.2 PME# (PCI Power Management Event)

The PME# signal comes from a PCI Express* device to request that the system be restarted. The PME# signal can generate an SMI#, SCI, or optionally a wake event. The event occurs when the PME# signal goes from high to low. No event is caused when it goes from low to high.

There is also an internal PME_BO_STS bit that will be set by the PCH when any internal device with PCI Power Management capabilities on bus 0 asserts the equivalent of the PME# signal. This is separate from the external PME# signal and can cause the same effect.

27.7.8.3 SYS_RESET# Signal

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

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

Note: The normal behavior for a SYS_RESET# assertion is host partition reset without power cycle. However, if bit 3 of the CF9h I/O register is set to '1' then SYS_RESET# will result in a full power-cycle reset.

Note: It is not recommended to use the PCH_PWROK pin for a reset button as it triggers a global power cycle reset.



Note: SYS_RESET# is in the primary power well but it only affects the system when PCH_PWROK is high.

27.7.8.4 THERMTRIP# Signal

If THERMTRIP# goes active, the processor is indicating an overheat condition, and the PCH immediately transitions to an S5 state, driving SLP_S3#, SLP_S4#, SLP_S5# low, and setting the GEN_PMCON_2.PTS bit. The transition will generally look like a power button override.

When a THERMTRIP# event occurs, the PCH will power down immediately without following the normal S0 -> S5 path. The PCH will immediately drive SLP_S3#, SLP_S4#, and SLP_S5# low within 1 us after sampling THERMTRIP# active.

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

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

PCH must only honor the THERMTRIP# pin while it is being driven to a valid state by the processor. The THERMTRIP# Valid Point = '0', implies PCH will start monitoring THERMTRIP# at PLTRST# de-assertion (default). The THERMTRIP# Valid Point = '1', implies PCH will start monitoring THERMTRIP# at CPUPWRGD assertion. Regardless of the setting, the PCH must stop monitoring THERMTRIP# at CPUPWRGD de-assertion.

Note: A thermal trip event will clear the PWRBTN_STS bit.

27.7.8.5 Sx_Exit_Holdoff#

When S3/S4/S5 is entered and SLP_A# is asserted, Sx_Exit_Holdoff# can be asserted by a platform component to delay resume to S0. SLP_A# de-assertion is an indication of the intent to resume to S0, but this will be delayed so long as Sx_Exit_Holdoff# is asserted. Sx_Exit_Holdoff is ignored outside of an S3/S4/S5 entry sequence with SLP_A# asserted. With the de-assertion of RSMRST# (either from G3->S0 or DeepSx->S0), this pin is a GPIO input and must be programmed by BIOS to operate as Sx_Exit_Holdoff. When SLP_A# is asserted (or it is de-asserted but Sx_Exit_Holdoff# is asserted), the PCH will not access SPI Flash. How a platform uses this signal is platform specific.

Requirements to support Sx_Exit_Holdoff#:

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

After the PCH has booted up to S0 at least once since the last G3 or DeepSx exit, the EC can begin monitoring SLP_A# and using the SX_EXIT_HOLDOFF# pin to stop the PCH from accessing flash. When SLP_A# asserts, if the EC intends to access flash, it will assert SX_EXIT_HOLDOFF#. To cover the case where the PCH is going through a



global reset, and not a graceful Sx+CMoff/Sx+CM3PG entry, the EC must monitor the SPI flash CS0# pin for 5ms after SLP_A# assertion before making the determination that it is safe to access flash.

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

27.7.9 ALT Access Mode

Before entering a low power state, several registers from powered down parts may need to be saved. In the majority of cases, this is not an issue, as registers have read and write paths. However, several of the ISA compatible registers are either read only or write only. To get data out of write-only registers, and to restore data into read-only registers, the PCH implements an ALT access mode.

If the ALT access mode is entered and exited after reading the registers of the PCH timer (8254), the timer starts counting faster (13.5 ms). The following steps listed below can cause problems:

1. BIOS enters ALT access mode for reading the PCH timer related registers.
2. BIOS exits ALT access mode.
3. BIOS continues through the execution of other needed steps and passes control to the operating system.

After getting control in step #3, if the operating system does not reprogram the system timer again, the timer ticks may be happening faster than expected.

Operating systems reprogram the system timer and therefore do not encounter this problem.

For other operating systems, the BIOS should restore the timer back to 54.6 ms before passing control to the operating system. If the BIOS is entering ALT access mode before entering the suspend state it is not necessary to restore the timer contents after the exit from ALT access mode.

27.7.9.1 Write Only Registers with Read Paths in ALT Access Mode

The registers described in [Table 27-11](#) have read paths in ALT access mode. The access number field in the table indicates which register will be returned per access to that port.



Table 27-11. Write Only Registers with Read Paths in ALT Access Mode

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
20h	12	1	PIC ICW2 of Master controller	40h	7	1	Timer Counter 0 status, bits [5:0]
		2	PIC ICW3 of Master controller			2	Timer Counter 0 base count low byte
		3	PIC ICW4 of Master controller			3	Timer Counter 0 base count high byte
		4	PIC OCW1 of Master controller ¹			6	Timer Counter 2 base count low byte
		5	PIC OCW2 of Master controller			7	Timer Counter 2 base count high byte
		6	PIC OCW3 of Master controller	42h	1	Timer Counter 2 status, bits [5:0]	
		7	PIC ICW2 of Slave controller	70h	1	Bit 7 = Read value is `0'. Bits [6:0] = RTC Address	
		8	PIC ICW3 of Slave controller				
		9	PIC ICW4 of Slave controller				
		10	PIC OCW1 of Slave controller ¹				
		11	PIC OCW2 of Slave controller				
		12	PIC OCW3 of Slave controller				

Notes:
 1. The OCW1 register must be read before entering ALT access mode.
 2. Bits 5, 3, 1, and 0 return 0.

27.7.9.2 PIC Reserved Bits

Many bits within the PIC are reserved, and must have certain values written in order for the PIC to operate properly. Therefore, there is no need to return these values in ALT access mode. When reading PIC registers from 20h and A0h, the reserved bits shall return the values listed in Table 27-12.

Table 27-12. PIC Reserved Bits Return Values

PIC Reserved Bits	Value Returned
ICW2(2:0)	000
ICW4(7:5)	000
ICW4(3:2)	00
ICW4(0)	0
OCW2(4:3)	00
OCW3(7)	0
OCW3(5)	Reflects bit 6
OCW3(4:3)	01

27.7.10 System Power Supplies, Planes, and Signals

27.7.10.1 Power Plane Control

The SLP_S3# output signal can be used to cut power to the system core supply, since it only goes active for the Suspend-to-RAM state (typically mapped to ACPI S3). Power must be maintained to the PCH primary well, and to any other circuits that need to



generate Wake signals from the Suspend-to-RAM state. During S3 (Suspend-to-RAM) all signals attached to powered down planes will be tri-stated or driven low, unless they are pulled using a Pull-up resistor.

Cutting power to the system core supply may be done using the power supply or by external FETs on the motherboard.

The SLP_S4# output signal is used to remove power to additional subsystems that are powered during SLP_S3#, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done using the power supply, or by external FETs on the motherboard.

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

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

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

27.7.10.2 SLP_S4# and Suspend-to-RAM Sequencing

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

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

27.7.10.3 PCH_PWROK Signal

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

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

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

27.7.10.4 BATLOW# (Battery Low)

The BATLOW# input can inhibit waking from S3, S4, S5 and Deep Sx states if there is not sufficient power. It also causes an SMI if the system is already in an S0 state.



27.7.10.5 SLP_LAN# Pin Behavior

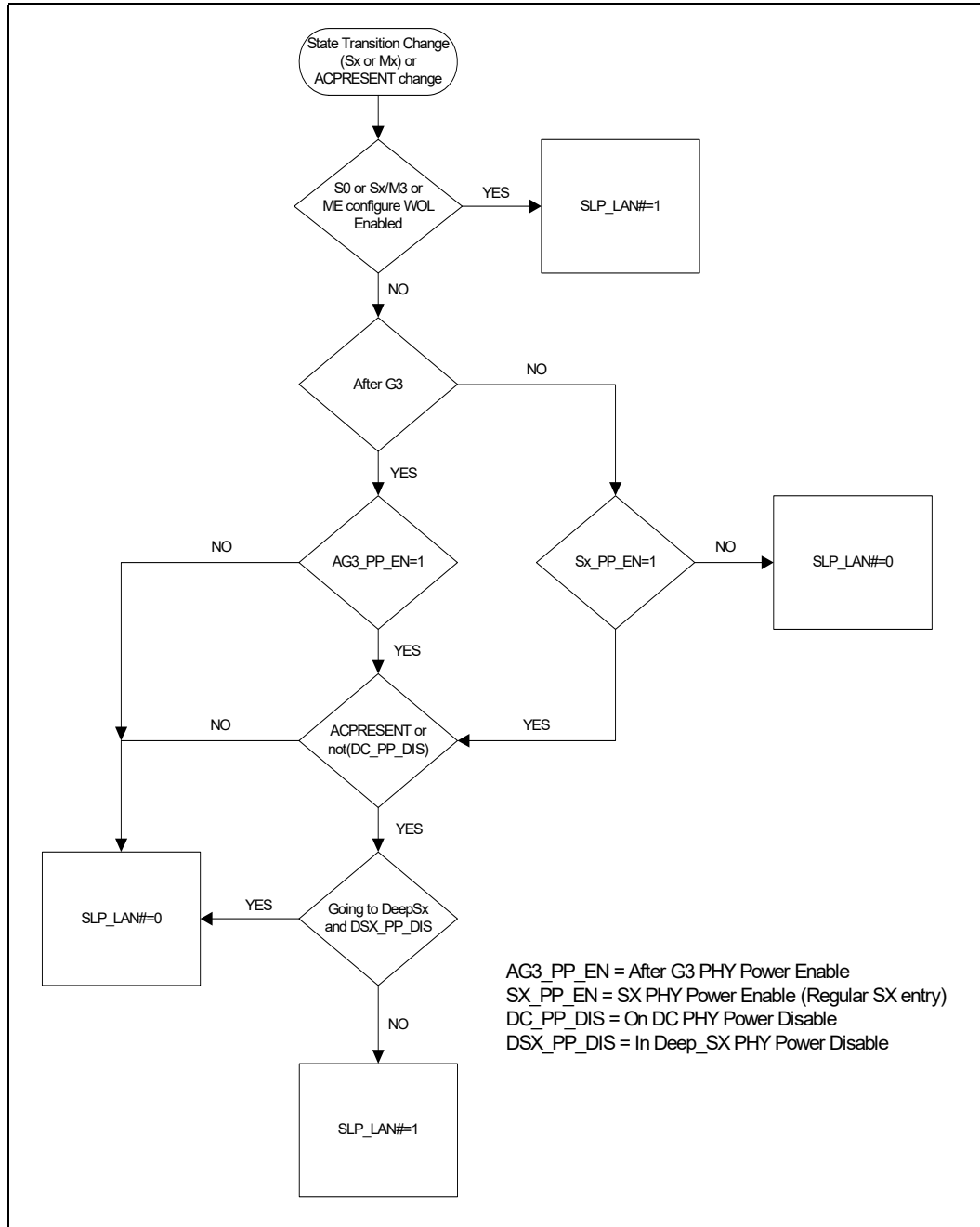
The PCH controls the voltage rails into the external LAN PHY using the SLP_LAN# pin.

- The LAN PHY is always powered when the Host and Intel® CSME systems are running.
 - SLP_LAN#='1' whenever SLP_S3#='1' or SLP_A#='1'.
- If the LAN PHY is required by Intel® CSME in Sx/M-Off or Deep Sx, Intel® CSME must configure SLP_LAN#='1' irrespective of the power source and the destination power state. Intel® CSME must be powered at least once after G3 to configure this.
- If the LAN PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN.
- If the LAN PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN.
- If the LAN PHY is required in Deep Sx, the host BIOS must keep DSX_PP_DIS cleared.
- If the LAN PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS.

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

The flow chart below shows how a decision is made to drive SLP_LAN# every time its policy needs to be evaluated.

Figure 27-1. Conceptual Diagram of SLP_LAN#





27.7.10.6 SLP_WLAN# Pin Behavior

The PCH controls the voltage rails into the external wireless LAN PHY using the SLP_WLAN# pin.

- The wireless LAN PHY is always powered when the Host is running.
 - SLP_WLAN#='1' whenever SLP_S3#='1'.
- If Wake on Wireless LAN (WoWLAN) is required from S3/S4/S5 states, the host BIOS must set HOST_WLAN_PP_EN.
- If WoWLAN is required from Deep Sx, the host BIOS must set DSX_WLAN_PP_EN.
- If Intel® CSME has access to the Wireless LAN device:
 - The Wireless LAN device must always be powered as long as Intel® CSME is powered. SLP_WLAN#='1' whenever SLP_A#='1'.
 - If Wake on Wireless LAN (WoWLAN) is required from M-Off state, Intel® CSME will configure SLP_WLAN#='1' in Sx/M-Off.

Intel® CSME configuration of SLP_WLAN# in Sx/M-Off is dependent on Intel® CSME power policy configuration. When the Wireless LAN device is an integrated connectivity device (CNVi) the power to the CNVi external RF chip (CRF) must be always on. In this case the SLP_WLAN# shall not control the CRF 3.3V power rail.

27.7.10.7 SUSPWRDNACK/SUSWARN#/GPP_A13 Steady State Pin Behavior

Table 27-13 summarizes SUSPWRDNACK/SUSWARN#/GPP_A13 pin behavior.

Table 27-13. SUSPWRDNACK/SUSWARN#/GPP_A13 Pin Behavior

Pin	Deep Sx (Supported /Not-Supported)	GPP_A13 Input/Output (Determine by GP_IO_SEL bit)	Pin Value in S0	Pin Value in Sx/M-Off	Pin Value in Sx/M3	Pin Value in Deep Sx
SUSPWRDNACK	Not Supported	Native	0	Depends on Intel® CSME power package and power source (Note 1)	0	Off
SUSWARN#	Supported	Native	1	1 (Note 2)	1	Off
GPP_A13	Do not Care	IN	High-Z	High-Z	High-Z	Off
	Do not Care	OUT	Depends on GPP_A13 output data value	Depends on GPP_A13 output data value	Depends on GPP_A13 output data value	Off
Notes: 1. PCH will drive SPDA pin based on Intel® CSME power policy configuration. 2. If entering Deep Sx, pin will assert and become undriven (Off) when suspend well drops upon Deep Sx entry.						

Table 27-14. SUSPWRDNACK During Reset (Sheet 1 of 2)

Reset Type (Note)	SPDA Value
power-cycle Reset	0
Global Reset	0



Table 27-14. SUSPWRDNACK During Reset (Sheet 2 of 2)

Reset Type (Note)	SPDA Value
Straight to S5	PCH initially drive '0' and then drive per Intel® CSME power policy configuration.
Note: Refer Table 27-15	

27.7.10.8 RTCRST# and SRTCST#

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

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

27.7.11 Legacy Power Management Theory of Operation

Instead of relying on ACPI software, legacy power management uses BIOS and various hardware mechanisms. The scheme relies on the concept of detecting when individual subsystems are idle, detecting when the whole system is idle, and detecting when accesses are attempted to idle subsystems.

However, the operating system is assumed to be at least APM enabled. Without APM calls, there is no quick way to know when the system is idle between keystrokes. The PCH does not support burst modes.

27.7.11.1 Mobile APM Power Management

In mobile systems, there are additional requirements associated with device power management. To handle this, the PCH has specific SMI traps available. The following algorithm is used:

1. The periodic SMI timer checks if a device is idle for the require time. If so, it puts the device into a low-power state and sets the associated SMI trap.
2. When software (not the SMI handler) attempts to access the device, a trap occurs (the cycle does not really go to the device and an SMI is generated).
3. The SMI handler turns on the device and turns off the trap.
4. The SMI handler exits with an I/O restart. This allows the original software to continue.

27.7.12 Reset Behavior

When a reset is triggered, the PCH will send a warning message to the processor to allow the processor to attempt to complete any outstanding memory cycles and put memory into a safe state before the platform is reset. When the processor is ready, it will send an acknowledge message to the PCH. Once the message is received the PCH asserts PLTRST#.



The PCH does not require an acknowledge message from the processor to trigger PLTRST#. A global reset will occur after 4 seconds if an acknowledge from the processor is not received.

When the PCH causes a reset by asserting PLTRST#, its output signals will go to their reset states.

A reset in which the host platform is reset and PLTRST# is asserted is called a Host Reset or Host Partition Reset. Depending on the trigger a host reset may also result in power cycling refer [Table 27-15](#) for details. If a host reset is triggered and the PCH times out before receiving an acknowledge message from the processor a Global Reset with power-cycle will occur.

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

Straight to S5 is another reset type where all power wells that are controlled by the SLP_S3#, SLP_S4#, and SLP_A# pins, as well as SLP_S5# and SLP_LAN# (if pins are not configured as GPIOs), are turned off. All PCH functionality is reset except RTC Power Well backed information and Suspend well status, configuration, and functional logic for controlling and reporting the reset. The host stays there until a valid wake event occurs.

[Table 27-15](#) shows the various reset triggers.

Table 27-15. Causes of Host and Global Resets (Sheet 1 of 2)

Trigger	Host Reset Without Power Cycle ¹	Host Reset With Power Cycle ²	Global Reset With Power Cycle ³	Straight to S5 ⁶ (Host Stays There)
Write of 0Eh to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=0b	No	Yes	No (Note 4)	
Write of 06h to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=0b	Yes	No	No (Note 4)	
Write of 06h or 0Eh to CF9h (RST_CNT Register) when CF9h when Global Reset Bit=1b	No	No	Yes	
SYS_RESET# Asserted and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No (Note 4)	
SYS_RESET# Asserted and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No (Note 4)	
SMBus Slave Message received for Reset with Power-Cycle	No	Yes	No (Note 4)	
SMBus Slave Message received for Reset without Power-Cycle	Yes	No	No (Note 4)	
SMBus Slave Message received for unconditional Power Down	No	No	No	Yes
TCO Watchdog Timer reaches zero two times	Yes	No	No (Note 4)	
Power Failure: PCH_PWROK signal goes inactive in S0 or DSW_PWROK drops	No	No	Yes	
SYS_PWROK Failure: SYS_PWROK signal goes inactive in S0	No	No	Yes	
Processor Thermal Trip (THERMTRIP#) causes transition to S5 and reset asserts	No	No	No	Yes



Table 27-15. Causes of Host and Global Resets (Sheet 2 of 2)

Trigger	Host Reset Without Power Cycle ¹	Host Reset With Power Cycle ²	Global Reset With Power Cycle ³	Straight to S5 ⁶ (Host Stays There)
PCH internal thermal sensors signals a catastrophic temperature condition	No	No	No	Yes
Power Button 4 second override causes transition to S5 and reset asserts	No	No	No	Yes
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 1	No	No	Yes	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No (Note 4)	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No (Note 4)	
Intel® Management Engine Triggered Host Reset without Power-Cycle	Yes	No	No (Note 4)	
Intel® Management Engine Triggered Host Reset with Power-Cycle	No	Yes	No (Note 4)	
Intel® Management Engine Triggered Power Button Override	No	No	No	Yes
Intel® Management Engine Watchdog Timer Timeout	No	No	No (Note 8)	Yes
Intel® Management Engine Triggered Global Reset	No	No	Yes	
Intel® Management Engine Triggered Host Reset with power down (host stays there)	No	Yes (Note 5)	No (Note 4)	
PLTRST# Entry Timeout (Note 7)	No	No	Yes	
CPUPWRGD Stuck Low	No	No	Yes	
Power Management Watchdog Timer	No	No	No (Note 8)	Yes
Intel® Management Engine Hardware Uncorrectable Error	No	No	No (Note 8)	Yes
<p>Notes:</p> <ol style="list-style-type: none"> 1. The PCH drops this type of reset request if received while the system is in S3/S4/S5. 2. PCH does not drop this type of reset request if received while system is in a software-entered S3/S4/S5 state. However, the PCH will perform the reset without executing the RESET_WARN protocol in these states. 3. The PCH does not send warning message to processor, reset occurs without delay. 4. Trigger will result in Global Reset with Power-Cycle if the acknowledge message is not received by PCH. 5. The PCH waits for enabled wake event to complete reset. 6. Upon entry to S5, if Deep Sx is enabled and conditions are met per Section 27.7.7.6, the system will transition to Deep Sx. 7. PLTRST# Entry Timeout is automatically initiated if the hardware detects that the PLTRST# sequence has not been completed within 4 seconds of being started. 8. Trigger will result in Global Reset with Power-Cycle if AGR_LS_EN=1 and Global Reset occurred while the current or destination state was S0. 				



28 Real Time Clock (RTC)

28.1 Acronyms

Acronyms	Description
ESR	Equivalent Series Resistance. Resistive element in a circuit such as a clock crystal.
GPI	General Purpose Input
PPM	Parts Per Million. Crystal accuracy or frequency variation indicator.
RAM	Random Access Memory

28.2 Overview

The PCH contains a Motorola* MC146818B-compatible real-time clock with 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. The RTC operates on a 32.768 kHz crystal and a 3V 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 30 days in advance, rather than just 24 hours in advance.

28.3 Signal Description

Name	Type	Description
RTCX1	I	Crystal Input 1: This signal is connected to the 32.768 kHz crystal. If no external crystal is used, then RTCX1 can be driven with the desired clock rate. Maximum voltage allowed on this pin is 1.2V.
RTCX2	O	Crystal Input 2: This signal is connected to the 32.768 kHz crystal. If no external crystal is used, then RTCX2 must be left floating.
RTCST#	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 RTCST# 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 RTCST# pin must rise before the DSW_PWROK pin.
SRTCST#	I	Secondary RTC Reset: This signal resets the manageability register bits in the RTC well when the RTC battery is removed. Notes: 1. The SRTCST# 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 SRTCST# pin must rise before the DSW_PWROK pin. 3. SRTCST# and RTCST# should not be shorted together.



28.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset*	Immediately after Reset*	S3/S4/S5	Deep Sx
RTCRST#	RTC	Undriven	Undriven	Undriven	Undriven
SRTCRST#	RTC	Undriven	Undriven	Undriven	Undriven
Note: *Reset reference for RTC well pins is RTCRST#.					

28.5 Functional Description

The Real Time Clock (RTC) module provides a battery backed-up date and time keeping device with two banks of static RAM with 128 bytes each, although the first bank has 114 bytes for general purpose usage.

Three interrupt features are available: time of day alarm with once a second to once a month range, periodic rates of 122 – 500 ms, and end of update cycle notification. Seconds, minutes, hours, days, day of week, month, and year are counted. Daylight savings compensation is no longer supported.

The hour is represented in twelve or twenty-four hour format, and data can be represented in BCD or binary format. The design is functionally compatible with the Motorola MS146818B. The time keeping comes from a 32.768 kHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block has very specific functions. The first ten are for time and date information. The next four (0Ah to 0Dh) are registers, which configure and report RTC functions.

The time and calendar data should match the data mode (BCD or binary) and hour mode (12 or 24 hour) as selected in register B. It is up to the programmer to make sure that data stored in these locations is within the reasonable values ranges and represents a possible date and time. The exception to these ranges is to store a value of C0–FFh in the Alarm bytes to indicate a do not care situation. All Alarm conditions must match to trigger an Alarm Flag, which could trigger an Alarm Interrupt if enabled.

The SET bit must be 1 while programming these locations to avoid clashes with an update cycle. Access to time and date information is done through the RAM locations. If a RAM read from the ten time and date bytes is attempted during an update cycle, the value read do not necessarily represent the true contents of those locations. Any RAM writes under the same conditions are ignored.

Note: The leap year determination for adding a 29th day to February does not take into account the end-of-the-century exceptions. The logic simply assumes that all years divisible by 4 are leap years. According to the Royal Observatory Greenwich, years that are divisible by 100 are typically not leap years. In every fourth century (years divisible by 400, like 2000), the 100-year-exception is over-ridden and a leap-year occurs.

Note: The year 2100 will be the first time in which the current RTC implementation would incorrectly calculate the leap-year.

The PCH does not implement month/year alarms.



28.5.1 Update Cycles

An update cycle occurs once a second, if the SET bit of register B is not asserted and the divide chain is properly configured. During this procedure, the stored time and date are incremented, overflow is checked, a matching alarm condition is checked, and the time and date are rewritten to the RAM locations.

The update cycle will start at least 488 μ s after the UIP bit of register A is asserted, and the entire cycle does not take more than 1984 μ s to complete. The time and date RAM locations (0–9) are disconnected from the external bus during this time.

To avoid update and data corruption conditions, external RAM access to these locations can safely occur at two times. When a update-ended interrupt is detected, almost 999 ms is available to read and write the valid time and date data. If the UIP bit of Register A is detected to be low, there is at least 488 μ s before the update cycle begins.

Warning: The overflow conditions for leap years adjustments are based on more than one date or time item. To ensure proper operation when adjusting the time, the new time and data values should be set at least two seconds before leap year occurs.

28.5.2 Interrupts

The real-time clock interrupt is internally routed within the PCH both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt does not leave the PCH, nor is it shared with any other interrupt. IRQ8# from the SERIRQ stream is ignored. However, the High Performance Event Timers can also be mapped to IRQ8#; in this case, the RTC interrupt is blocked.

28.5.3 Lockable RAM Ranges

The RTC battery-backed RAM supports two 8-byte ranges that can be locked using the configuration space. If the locking bits are set, the corresponding range in the RAM will not be readable or writable. A write cycle to those locations will have no effect. A read cycle to those locations will not return the location's actual value (resultant value is undefined).

Once a range is locked, the range can be unlocked only by a hard reset, which will invoke the BIOS and allow it to relock the RAM range.

28.5.4 Century Rollover

The PCH detects a rollover when the Year byte transitions from 99 to 00. Upon detecting the rollover, the PCH sets the NEWCENTURY_STS bit.

If the system is in an S0 state, this causes an SMI#. The SMI# handler can update registers in the RTC RAM that are associated with century value.

If the system is in a sleep state (S3–S5) when the century rollover occurs, the PCH also sets the NEWCENTURY_STS bit, but no SMI# is generated. When the system resumes from the sleep state, BIOS should check the NEWCENTURY_STS bit and update the century value in the RTC RAM.



28.5.5 Clearing Battery-Backed RTC RAM

Clearing CMOS RAM in a PCH-based platform can be done by using a jumper on RTCRST# or GPI. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low.

28.5.5.1 Using RTCRST# to Clear CMOS

A jumper on RTCRST# can be used to clear CMOS values, as well as reset to default, the state of those configuration bits that reside in the RTC power well.

When the RTCRST# is strapped to ground, the RTC_PWR_STS bit will be set and those configuration bits in the RTC power well will be set to their default state. BIOS can monitor the state of this bit and manually clear the RTC CMOS array once the system is booted. The normal position would cause RTCRST# to be pulled up through a weak Pull-up resistor. This RTCRST# jumper technique allows the jumper to be moved and then replaced—all while the system is powered off. Then, once booted, the RTC_PWR_STS can be detected in the set state.

28.5.5.2 Using a GPI to Clear CMOS

A jumper on a GPI can also be used to clear CMOS values. BIOS would detect the setting of this GPI on system boot-up, and manually clear the CMOS array.

Note: The GPI strap technique to clear CMOS requires multiple steps to implement. The system is booted with the jumper in new position, then powered back down. The jumper is replaced back to the normal position, then the system is rebooted again.

Warning: Do not implement a jumper on VccRTC to clear CMOS.

28.5.6 External RTC Circuitry

The PCH implements an internal oscillator circuit that is sensitive to step voltage changes in VCCRTC.

Table 28-1. RTC Crystal Requirements

Parameter	Specification
Frequency	32.768 kHz
Typical Tolerance	20 ppm or better
ESR	≤ 50 kohm

Table 28-2. External Crystal Oscillator Requirements

Parameter	Specification
Frequency	32.768 kHz
Typical Tolerance	20 ppm or better
Voltage Swing	0 to 1.0Vp-p (±5%)





29 Serial ATA (SATA)

29.1 Acronyms

Acronyms	Description
AHCI	Advanced Host Controller Interface
DMA	Direct Memory Access
DEVSLP	Device Sleep
IDE	Integrated Drive Electronics
RAID	Redundant Array of Independent Disks
SATA	Serial Advanced Technology Attachment

29.2 References

Specification	Location
Serial ATA Specification, Revision 3.2	https://www.sata-io.org
Serial ATA II: Extensions to Serial ATA 1.0, Revision 1.0	https://www.sata-io.org
Serial ATA II Cables and Connectors Volume 2 Gold	https://www.sata-io.org
Advanced Host Controller Interface Specification	http://www.intel.com/content/www/us/en/io/serial-ata/ahci.html

29.3 Overview

The PCH SATA controller support two modes of operation, AHCI mode using memory space and RAID mode. The PCH 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.

29.4 Signals Description

Table 29-1. Signals Description (Sheet 1 of 5)

Name	Type	Description
SATA_DEVSLP0/GPP_E4	OD	<p>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. Design Constraint: No external Pull-up or Pull-down termination required when used as DEVSLP.</p> <p>Note: This pin can be mapped to SATA Port 0.</p>
SATA_DEVSLP1/GPP_E5	OD	<p>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. Design Constraint: No external Pull-up or Pull-down termination required when used as DEVSLP.</p> <p>Note: This pin can be mapped to SATA Port 1.</p>

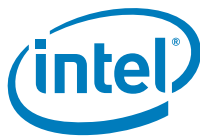


Table 29-1. Signals Description (Sheet 2 of 5)

Name	Type	Description
SATA_DEVSLP 2/GPP_E6	OD	Serial ATA Port [2] 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. Design Constraint: No external Pull-up or Pull-down termination required when used as DEVSLP. Note: This pin can be mapped to SATA Port 2.
SATA_DEVSLP 3/GPP_F5	OD	Serial ATA Port [3] 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. Design Constraint: No external pull-up or pull-down termination required when used as DEVSLP. Note: This pin can be mapped to SATA Port 3.
SATA_DEVSLP 4/GPP_F6	OD	Serial ATA Port [4] 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. Design Constraint: No external pull-up or pull-down termination required when used as DEVSLP. Note: This pin can be mapped to SATA Port 4.
SATA_DEVSLP 5/GPP_F7	OD	Serial ATA Port [5] 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. Design Constraint: No external pull-up or pull-down termination required when used as DEVSLP. Note: This pin can be mapped to SATA Port 5.
SATA_DEVSLP 7/GPP_F9	OD	Serial ATA Port [7] 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. Design Constraint: No external pull-up or pull-down termination required when used as DEVSLP. Note: This is applicable to Server/Workstation only. Note: This pin can be mapped to SATA Port 7.
SATA0A_TXP/PCIE11_TXP SATA0A_TXN/PCIE11_TXN	O	Serial ATA Differential Transmit Pair 0 [First Instance]: 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 11 signals.
SATA0A_RXP/PCIE11_RXP SATA0A_RXN/PCIE11_RXN	I	Serial ATA Differential Receive Pair 0 [First Instance]: 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 11 signals.
SATA1A_TXP/PCIE12_TXP SATA1A_TXN/PCIE12_TXN	O	Serial ATA Differential Transmit Pair 1 [First Instance]: 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 12.
SATA1A_RXP/PCIE12_RXP SATA1A_RXN/PCIE12_RXN	I	Serial ATA Differential Receive Pair 1 [First Instance]: 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 12.
SATA0B_TXP/PCIE13_TXP SATA0B_TXN/PCIE13_TXN	O	Serial ATA Differential Transmit Pair 0 [Second Instance]: 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 13 signals.



Table 29-1. Signals Description (Sheet 3 of 5)

Name	Type	Description
SATA0B_RXP/ PCIE13_RXP SATA0B_RXN/ PCIE13_RXN	I	Serial ATA Differential Receive Pair 0 [Second Instance]: 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 13 signals.
SATA1B_TXP/ PCIE14_TXP SATA1B_TXN/ PCIE14_TXN	O	Serial ATA Differential Transmit Pair 1 [Second Instance]: 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 11 signals.
SATA1B_RXP/ PCIE14_RXP SATA1B_RXN/ PCIE14_RXN	I	Serial ATA Differential Receive Pair 1 [Second Instance]: 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 11 signals.
SATA2_TXP/ PCIE15_TXP SATA2_TXN/ PCIE15_TXN	O	Serial ATA Differential Transmit Pair 2: These outbound SATA Port 2 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 15 signals.
SATA2_RXP/ PCIE15_RXP SATA2_RXN/ PCIE15_RXN	I	Serial ATA Differential Receive Pair 2: These inbound SATA Port 2 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 15 signals.
SATA3_TXP/ PCIE16_TXP SATA3_TXN/ PCIE16_TXN	O	Serial ATA Differential Transmit Pair 3: These outbound SATA Port 3 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 16 signals.
SATA3_RXP/ PCIE16_RXP SATA3_RXN/ PCIE16_RXN	I	Serial ATA Differential Receive Pair 3: These inbound SATA Port 3 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 16 signals.
SATA4_TXP/ PCIE17_TXP SATA4_TXN/ PCIE17_TXN	O	Serial ATA Differential Transmit Pair 4: These outbound SATA Port 4 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 17 signals.
SATA4_RXP/ PCIE17_RXP SATA4_RXN/ PCIE17_RXN	I	Serial ATA Differential Receive Pair 4: These inbound SATA Port 4 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 17 signals.
SATA5_TXP/ PCIE18_TXP SATA5_TXN/ PCIE18_TXN	O	Serial ATA Differential Transmit Pair 5: These outbound SATA Port 5 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 18 signals.
SATA5_RXP/ PCIE18_RXP SATA5_RXN/ PCIE18_RXN	I	Serial ATA Differential Receive Pair 5: These inbound SATA Port 5 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 18 signals.
SATA6_TXP/ PCIE19_TXP SATA6_TXN/ PCIE19_TXN	O	Serial ATA Differential Transmit Pair 6: These outbound SATA Port 6 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 19 signals. Note: This is applicable to Server/Workstation only.
SATA6_RXP/ PCIE19_RXP SATA6_RXN/ PCIE19_RXN	I	Serial ATA Differential Receive Pair 6: These inbound SATA Port 6 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 19 signals. Note: This is applicable to Server/Workstation only.
SATA7_TXP/ PCIE20_TXP SATA7_TXN/ PCIE20_TXN	O	Serial ATA Differential Transmit Pair 7: These outbound SATA Port 7 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 20 signals. Note: This is applicable to Server/Workstation only.



Table 29-1. Signals Description (Sheet 4 of 5)

Name	Type	Description
SATA7_RXP/ PCIE20_RXP SATA7_RXN/ PCIE20_RXN	I	Serial ATA Differential Receive Pair 7: These inbound SATA Port 7 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe* Port 20 signals. Note: This is applicable to Server/Workstation only.
SATAGP0/ GPP_E0 / SATAXPCIE0	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. Note: The default use of this pin is GPP_E0. Pin defaults to Native mode as SATAXPCIE0 depends on soft-strap.
SATAGP1/ GPP_E1 / SATAXPCIE1	I	Serial ATA Port [1] General Purpose Inputs: When configured as SATAGP1, this is an input pin that is used as an interlock switch status indicator for SATA Port 1. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Note: This default use of this pin is GPP_E1. Pin defaults to Native mode as SATAXPCIE1 depends on soft-strap.
SATAGP2/ GPP_E2 / SATAXPCIE2	I	Serial ATA Port [2] General Purpose Inputs: When configured as SATAGP2, this is an input pin that is used as an interlock switch status indicator for SATA Port 2. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Note: The default use of this pin is GPP_E2. Pin defaults to Native mode as SATAXPCIE2 depends on soft-strap.
SATAGP3/ SATAXPCIE3/ GPP_F0	I	Serial ATA Port [3] General Purpose Inputs: When configured as SATAGP3, this is an input pin that is used as an interlock switch status indicator for SATA Port 3. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Note: The default use of this pin is GPP_F0. Pin defaults to Native mode as SATAXPCIE3 depends on soft-strap.
SATAGP4/ SATAXPCIE4/ GPP_F1	I	Serial ATA Port [4] General Purpose Inputs: When configured as SATAGP4, this is an input pin that is used as an interlock switch status indicator for SATA Port 4. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Note: The default use of this pin is GPP_F1. Pin defaults to Native mode as SATAXPCIE4 depends on soft-strap.
SATAGP5/ SATAXPCIE5/ GPP_F2	I	Serial ATA Port [5] General Purpose Inputs: When configured as SATAGP5, this is an input pin that is used as an interlock switch status indicator for SATA Port 5. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Note: The default use of this pin is GPP_F2. Pin defaults to Native mode as SATAXPCIE5 depends on soft-strap.
SATALED#/ GPP_E8 / SPI1_CS1#	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. Note: An external Pull-up resistor to VCC3_3 is required.
SCLOCK/ GPP_F10	OD	SGPIO Reference Clock: The SATA controller uses rising edges of this clock to transmit serial data, and the target uses the falling edge of this clock to latch data. The SClock frequency supported is 32 kHz. If SGPIO interface is not used, this signal can be used as GPP_F10.
SLOAD/ GPP_F11	OD	SGPIO Load: The controller drives a '1' at the rising edge of SCLOCK to indicate either the start or end of a bit stream. A 4-bit vendor specific pattern will be transmitted right after the signal assertion. If SGPIO interface is not used, this signal can be used as GPP_F11.
SDATAOUT0/ GPP_F13	OD	SGPIO Dataout0: Driven by the controller to indicate the drive status in the following sequence: drive 0, 1,2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_F13.



Table 29-1. Signals Description (Sheet 5 of 5)

Name	Type	Description
SDATAOUT1/ GPP_F12	OD	SGPIO Dataout1: Driven by the controller to indicate the drive status in the following sequence: drive 0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_F12.

29.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor type	Notes
SATAXPCE[7:0]	Internal pull-up	Internal Pull-Up Resistors are 14k-26k unless specified.

29.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ³	Immediately after Reset ³	S3/S4/S5	Deep Sx
SATA0A_TXP/N, SATA0A_RXP/N	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATA0B_TXP/N, SATA0B_RXP/N	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATA1A_TXP/N, SATA1A_RXP/N	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATA1B_TXP/N, SATA1B_RXP/N	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATA[7:2]_TXP/N, SATA[7:2]_RXP/N ³	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
SATALED#/GPP_E81	Primary	Undriven	Undriven	Undriven	OFF
DEVSLP[2:0]/ GPP_E[6:4]1	Primary	Undriven	Undriven	Undriven	OFF
DEVSLP[7:3]/ GPP_F[9:5]1,3	Primary	Undriven	Undriven	Undriven	OFF
SATAGP[2:0]/ GPP_E[2:0]2	Primary	Undriven	Undriven	Undriven	OFF
SATAGP[7:3]/ GPP_F[4:0]2,3	Primary	Undriven	Undriven	Undriven	OFF
SATAXPCE[7:0]2,3	Primary	Internal Pull-up	Internal Pull-up	Undriven	OFF
SCLOCK/GPP_F10 ¹	Primary	Undriven	Undriven	Undriven	OFF
SLOAD/GPP_F11 ¹	Primary	Undriven	Undriven	Undriven	OFF
SDATAOUT0/ GPP_F13 ¹	Primary	Undriven	Undriven	Undriven	OFF
SDATAOUT1/ GPP_F12 ¹	Primary	Undriven	Undriven	Undriven	OFF

Notes:

- 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.
- Pin defaults to Native mode as SATAXPCE_x depends on soft-strap. SATA[7:6]_TXP/N, SATA[7:6]_RXP/N, DEVSLP[7:6]/GPP_F[9:8], SATAGP[7:6]/GPP_F[4:3], SATAXPCE[7:6] are for Server/Workstation only.
- Reset reference for primary well pins is RSMRST#.



29.7 Functional Description

The PCH SATA host controller (D23:F0) supports AHCI or RAID mode.

The PCH SATA controller does not support legacy IDE mode or combination mode.

The PCH SATA controller interacts with an attached mass storage device through a register interface that is compatible with an SATA AHCI/RAID host adapter. The host software follows existing standards and conventions when accessing the register interface and follows standard command protocol conventions.

29.7.1 SATA 6 Gb/s Support

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

29.7.2 SATA Feature Support

The PCH SATA controller is capable of supporting all AHCI 1.3 and AHCI 1.3.1, refer to the Intel web site on Advanced Host Controller Interface Specification for current specification status: <http://www.intel.com/content/www/us/en/io/serial-ata/ahci.html>.

For capability details, refer to PCH SATA controller register (D23:F0:Offset 00h CAP, and AHCI BAR PxCMD Offset 18h).

The PCH SATA controller does **not** support:

- Port Multiplier
- FIS Based Switching
- Command Based Switching
- IDE mode or combination mode
- Cold Presence Detect
- Function Level Reset (FLR)

29.7.3 Hot-Plug Operation

The PCH SATA controller supports Hot-Plug Surprise removal and Insertion Notification. An internal SATA port with a Mechanical Presence Switch can support PARTIAL and SLUMBER with Hot-Plug Enabled. Software can take advantage of power savings in the low power states while enabling Hot-Plug operation. Refer to Chapter 7 of the AHCI specification for details.

29.7.4 Intel® Rapid Storage Technology (Intel® RST)

The PCH SATA controller provides support for Intel® Rapid Storage Technology, providing both AHCI and integrated RAID functionality. The RAID capability provides high-performance/data-redundancy RAID 0/1/5/10 functionality on up to six ports for the PCH of the PCH SATA controller. Matrix RAID support is provided to allow multiple RAID levels to be combined on a single set of hard drives, such as RAID 0 and RAID 1 on two disks. Other RAID features include hot spare support, SMART alerting, and RAID 0 auto replace. Software components include an Option ROM and UEFI Driver for pre-



boot configuration and boot functionality, a Windows* compatible driver, and a user interface for configuration and management of the RAID capability of PCH SATA controller.

29.7.4.1 Intel® Rapid Storage Technology (Intel® RST) Configuration

Intel® RST offers several diverse options for RAID (redundant array of independent disks) to meet the needs of the end user. AHCI support provides higher performance and alleviates disk bottlenecks by taking advantage of the independent DMA engines that each SATA port offers in the PCH SATA controller.

- RAID Level 0 performance scaling up to 6 drives, enabling higher throughput for data intensive applications such as video editing.
- Data redundancy is offered through RAID Level 1, which performs mirroring.
- RAID Level 10 provides high levels of storage performance with data protection, combining the fault-tolerance of RAID Level 1 with the performance of RAID Level 0. By striping RAID Level 1 segments, high I/O rates can be achieved on systems that require both performance and fault-tolerance. RAID Level 10 requires 4 hard drives, and provides the capacity of two drives.
- RAID Level 5 provides highly efficient storage while maintaining fault-tolerance on 3 or more drives. By striping parity, and rotating it across all disks, fault tolerance of any single drive is achieved while only consuming 1 drive worth of capacity. That is, a 3-drive RAID 5 has the capacity of 2 drives, or a 4-drive RAID 5 has the capacity of 3 drives. RAID 5 has high read transaction rates, with a medium write rate. RAID 5 is well suited for applications that require high amounts of storage while maintaining fault tolerance.

By using the PCH's built-in Intel® Rapid Storage Technology, there is no loss of additional PCIe*/system resources or add-in card slot/motherboard space footprint used compared to when a discrete RAID controller is implemented. Intel® Rapid Storage Technology functionality requires the following items:

1. PCH SKU enabled for Intel® Rapid Storage Technology.
 - Intel® Rapid Storage Technology RAID Option ROM or UEFI Driver must be on the platform.
 - Intel® Rapid Storage Technology drivers, most recent revision.
 - At least two SATA hard disk drives (minimum depends on RAID configuration).

Intel® Rapid Storage Technology is not available in the following configurations:

- The SATA controller is programmed in RAID mode, but the AIE bit (D23:F0:Offset 9Ch bit 7) is set to 1.

29.7.4.2 Intel® Rapid Storage Technology (Intel® RST) RAID Option ROM

The Intel® Rapid Storage Technology RAID Option ROM is a standard PnP Option ROM that is easily integrated into any System BIOS. When in place, it provides the following three primary functions:

- Provides a text mode user interface that allows the user to manage the RAID configuration on the system in a pre-operating system environment. Its feature set is kept simple to keep size to a minimum, but allows the user to create and delete RAID volumes and select recovery options when problems occur.



- Provides boot support when using a RAID volume as a boot disk. It does this by providing Int13 services when a RAID volume needs to be accessed by MS-DOS applications (such as NTLDR) and by exporting the RAID volumes to the System BIOS for selection in the boot order.
- At each boot up, provides the user with a status of the RAID volumes and the option to enter the user interface by pressing CTRL-I.

29.7.5 Intel® Rapid Storage Technology enterprise (Intel® RSTe) - for Server/Workstation Only

The PCH SATA controller provides support for Intel® Rapid Storage Technology enterprise (RSTe), providing both AHCI and integrated RAID functionality. The RAID capability provides high-performance/data-redundancy RAID 0/1/5/10 functionality for the PCH of the PCH SATA controller. Matrix RAID support is provided to allow multiple RAID levels to be combined on a single set of hard drives, such as RAID 0 and RAID 1 on two disks. Other RAID features include hot spare support, rebuild on hot insert, SMART alerting, TRIM and SGPIO. Software components include a legacy Option ROM and UEFI Driver for pre-boot configuration and boot functionality, a Microsoft* Windows* compatible driver, and a user interface for configuration and management of the RAID capability of PCH SATA controller.

29.7.5.1 Intel® Rapid Storage Technology enterprise (Intel® RSTe) Configuration - for Server/Workstation Only

Intel® RSTe offers several diverse options for RAID (redundant array of independent disks) to meet the needs of the end user. AHCI support provides higher performance and alleviates disk bottlenecks by taking advantage of the independent DMA engines that each SATA port offers in the PCH SATA controller.

- RAID Level 0 performance scaling up to 8 drives, enabling higher throughput for data intensive applications such as video editing.
- Data redundancy is offered through RAID Level 1, which performs mirroring.
- RAID Level 10 provides high levels of storage performance with data protection, combining the fault-tolerance of RAID Level 1 with the performance of RAID Level 0. By striping RAID Level 1 segments, high I/O rates can be achieved on systems that require both performance and fault-tolerance. RAID Level 10 requires 4 hard drives, and provides the capacity of two drives.
- RAID Level 5 provides highly efficient storage while maintaining fault-tolerance on 3 or more drives. By striping parity, and rotating it across all disks, fault tolerance of any single drive is achieved while only consuming 1 drive worth of capacity. That is, a 3-drive RAID 5 has the capacity of 2 drives, or a 4-drive RAID 5 has the capacity of 3 drives. RAID 5 has high read transaction rates, with a medium write rate. RAID 5 is well suited for applications that require high amounts of storage while maintaining fault tolerance.

By using the PCH's built-in Intel® Rapid Storage Technology enterprise, there is no loss of additional PCIe*/system resources or add-in card slot/motherboard space footprint used compared to when a discrete RAID controller is implemented. Intel® Rapid Storage Technology enterprise functionality requires the following items:

1. PCH SKU enabled for Intel® Rapid Storage Technology enterprise.
2. Intel® Rapid Storage Technology enterprise RAID Option ROM or UEFI Driver must be on the platform.



3. Intel® Rapid Storage Technology enterprise drivers, most recent revision.
4. At least two SATA hard disk drives (minimum depends on RAID configuration).

Intel® Rapid Storage Technology enterprise is only available in the following configurations:

1. The SATA controller is programmed in RAID mode, but the AIE bit (D23:F0:Offset 9Ch bit 7) is set to 0.

29.7.5.2 Intel® Rapid Storage Technology enterprise (Intel® RSTe) Legacy RAID Option ROM - for Server/Workstation Only

The Intel® Rapid Storage Technology enterprise Legacy RAID Option ROM is a standard PnP Option ROM that is easily integrated into any System BIOS. When in place, it provides the following three primary functions:

- Provides a text mode user interface that allows the user to manage the RAID configuration on the system in a pre-operating system environment. Its feature set is kept simple to keep size to a minimum, but allows the user to create and delete RAID volumes and select recovery options when problems occur.
- Provides boot support when using a RAID volume as a boot disk. It does this by providing Int13 services when a RAID volume needs to be accessed by MS-DOS applications (such as NTLDR) and by exporting the RAID volumes to the System BIOS for selection in the boot order.
- At each boot up, provides the user with a status of the RAID volumes and the option to enter the user interface.

29.7.5.3 Intel® Rapid Storage Technology enterprise (Intel® RSTe) EFI Driver - for Server/Workstation Only

The Intel® Rapid Storage Technology enterprise EFI driver is a standard EFI driver that is easily integrated into any System BIOS. When in place, it provides the following three primary functions:

- Provides a text mode user interface (standard HII interface) that allows the user to manage the RAID configuration on the system in a pre-operating system environment. Its feature set is kept simple to keep size to a minimum, but allows the user to create and delete RAID volumes and select recovery options when problems occur.
- Provides boot support when using a RAID volume as a boot disk.

29.7.6 Power Management Operation

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

29.7.6.1 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.

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.

29.7.6.2 Power State Transitions

29.7.6.2.1 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.

29.7.6.2.2 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.

29.7.6.2.3 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 than sending commands over the interface to the device. The command most likely to be used in ATA/ATAPI is the "STANDBY IMMEDIATE" command.

29.7.6.2.4 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.

29.7.6.3 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.

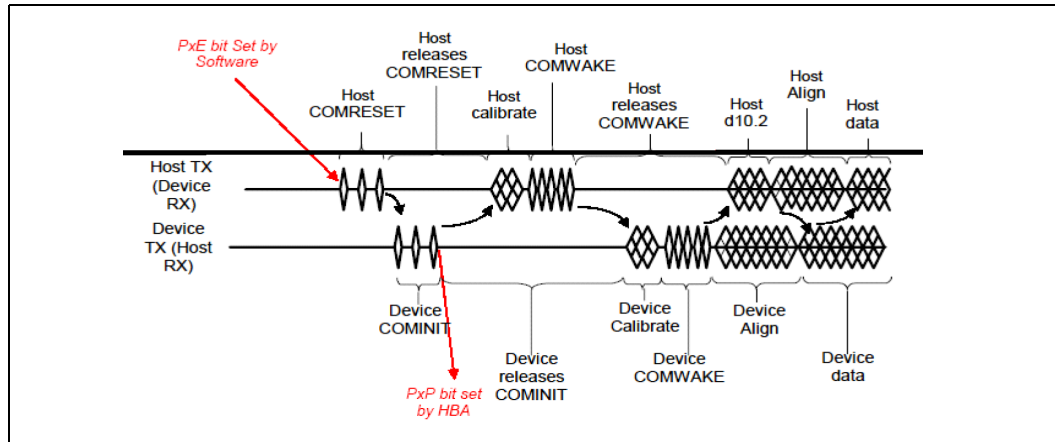
The SATA MPHY Dynamic Power Gating (PHYDPGEPx) can be enabled/disabled for each SATA ports. Refer to SATA SIR Index 90h for the PHYDPGEPx register details.

29.7.7 SATA Device Presence

The flow used to indicate SATA device presence is shown in [Figure 29-1](#). The 'PxE' bit refers to PCS.P[7:0]E bits, depending on the port being checked and the 'PxP' bits refer to the PCS.P[2:0]P 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 view if a new device is connected.

Figure 29-1. Flow for Port Enable/Device Present Bits



29.7.8 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.

29.7.9 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. Refer Section 7.3.1 of the *AHCI Specification* for more information.

29.7.10 Enclosure Management (SGPIO Signals)

Enclosure management is a mechanism by which the storage driver can monitor and control auxiliary service in a drive enclosure. This feature is only valid in AHCI/RAID mode.

The SGPIO signals are used in the enclosure management protocol (refer to SFF-8485 specification) and supports multiple-activity LEDs to show the per drive status information.

Note: These signals are not related to SATALED#.

The SGPIO group interfaces with an external controller chip that fetches and serializes the data for driving across the SGPIO bus. The output signals then control the LEDs within the enclosure. The PCH SATA controller only supports LED message transmission and has three SGPIO protocol signals implemented, that is SCLOCK, SDATAOUT and SLOAD.

Note: Intel does not validate all possible usage cases of this feature. Customers should validate their specific design implementation on their own platforms.

29.7.10.1 Mechanism

The enclosure management for SATA Controller involves sending messages that control LEDs in the enclosure. The messages for this function are stored after the normal registers in the AHCI BAR, at Offset 580h bytes for the PCH from the beginning of the AHCI BAR as specified by the EM_LOC global register.

Software creates messages for transmission in the enclosure management message buffer. The data in the message buffer should not be changed if CTL.TM bit is set by software to transmit an update message. Software should only update the message buffer when CTL.TM bit is cleared by hardware otherwise the message transmitted will be indeterminate. Software then writes a register to cause hardware to transmit the message or take appropriate action based on the message content. The software should only create message types supported by the controller, which is LED messages for the PCH. If the software creates other non LED message types (such as, SAF-TE, SES-2), the SGPIO interface may hang and the result is indeterminate.

During reset all SGPIO pins will be in tri-state. The interface will continue staying in tri-state after reset until the first transmission occurs, when software programs the message buffer and sets the transmit bit CTL.TM. The SATA host controller will initiate the transmission by driving SCLOCK and at the same time driving the SLOAD to "0" prior to the actual bit stream transmission. The Host will drive SLOAD low for at least 5 SCLOCK then only start the bit stream by driving the SLOAD to high. SLOAD will be driven high for 1 SCLOCK, followed by vendor-specific pattern that is default to "0000" if software is yet to program the value. A total of 24-bit streams from 8 ports (Port 0, Port 1, Port 2, Port 3, Port 4, Port 5, Port 6, Port 7) of 3-bit per port LED message will be transmitted on SDATAOUT0 pin after the SLOAD is driven high for 1 SCLOCK. For 8 SATA port configuration, only 4 ports (port 4, port 5, port 6 and port 7) of 12 bit total LED message follow by 12 bits of tri-state value will be transmitted out on SDATAOUT1 pin. For 6 SATA port configuration, only 2 ports (port 4 and port 5) of 6 bit total LED



message follow by 18 bits of tri-state value will be transmitted out on SDATAOUT1 pin. For 4 SATA port configuration, SDATAOUT1 pin is not required hence can be tri-state always.

All the default LED message values will be high prior to software setting them, except the Activity LED message that is configured to be hardware driven that will be generated based on the activity from the respective port. All the LED message values will be driven to '1' for the port that is unimplemented as indicated in the Port Implemented register regardless of the software programmed value through the message buffer.

There are 2 different ways of resetting the PCH's SGPIO interface, asynchronous reset and synchronous reset. Asynchronous reset is caused by platform reset to cause the SGPIO interface to be tri-state asynchronously. Synchronous reset is caused by setting the CTL.RESET bit, or HBA reset, where Host Controller will complete the existing full bit stream transmission then only tri-state all the SGPIO pins. After the reset, both synchronous reset and asynchronous reset, the SGPIO pins will stay tri-stated.

Note: The PCH Host Controller does not ensure that it will cause the target SGPIO device or controller to be reset. Software is responsible to keep the PCH SGPIO interface in tri-state for 2 second to cause a reset on the target of the SGPIO interface.

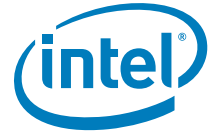
29.7.10.2 Message Format

Messages shall be constructed with a one DWord header that describes the message to be sent followed by the actual message contents. The first DWord shall be constructed as shown in Enclosure Management Message Format (EM_MF) register, refer to PCH Datasheet Volume 2.

The SAF-TE, SES-2, and SGPIO message formats are defined in the corresponding specifications, respectively. The LED message type is defined in the Enclosure Management LED (EM_LED) register, refer to PCH Datasheet Volume 2. It is the responsibility of software to ensure the content of the message format is correct. If the message type is not programmed as 'LED' for this controller, the controller shall not take any action to update its LEDs. For LED message type, the message size always consists of 4 bytes.

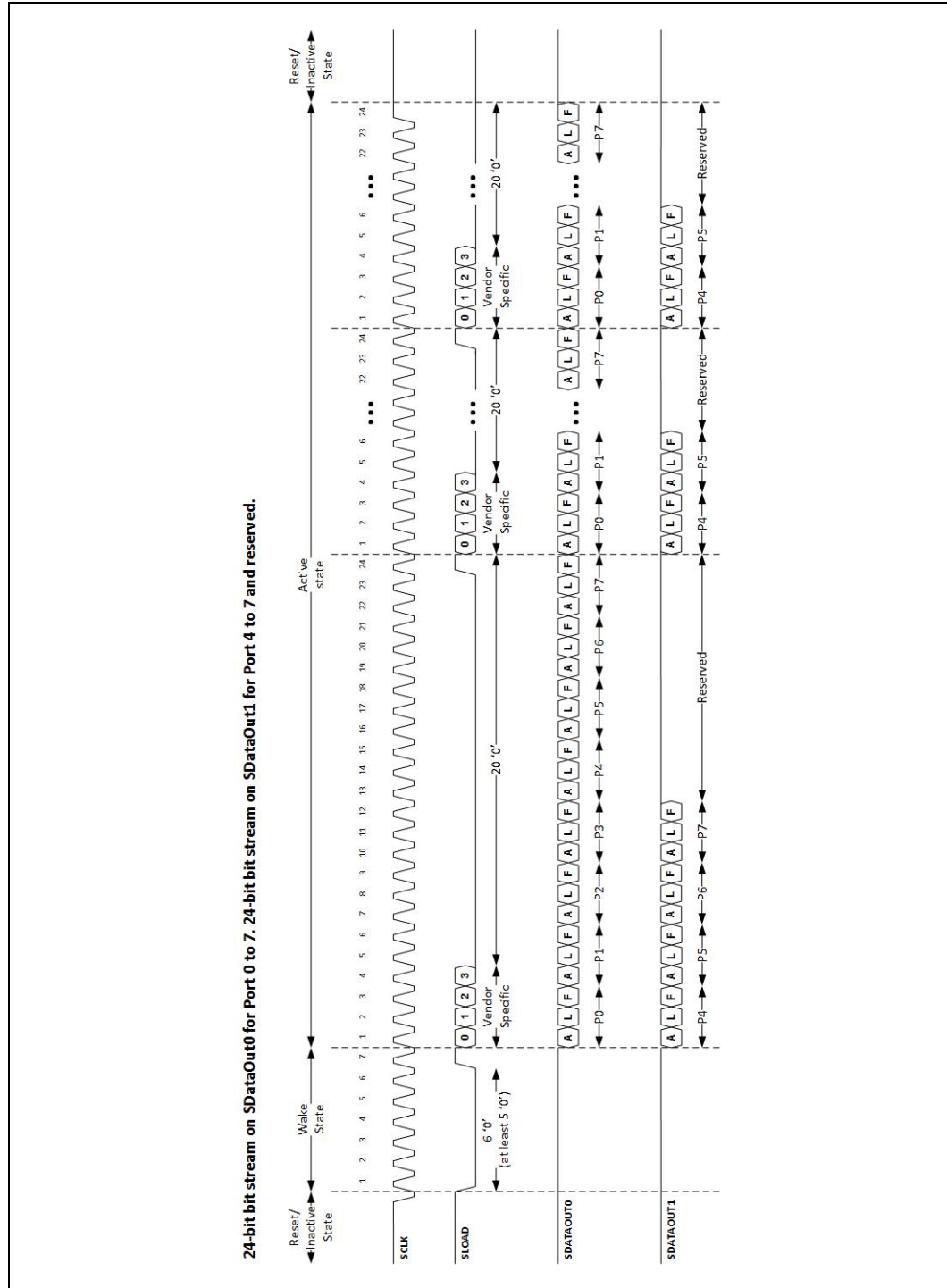
29.7.10.3 LED Message Type

The LED message type specifies the status of up to three LEDs. Typically, the usage for these LEDs is activity, fault, and locate. Not all implementations necessarily contain all LEDs (for example, some implementations may not have a locate LED). The message identifies the HBA port number and the Port Multiplier port number that the slot status applies to. If a Port Multiplier is not in use with a particular device, the Port Multiplier port number shall be '0'. The format of the LED message type is defined in the Enclosure Management LED (EM_LED) register, refer to PCH Datasheet Volume 2. The LEDs shall retain their values until there is a following update for that particular slot.



29.7.10.4 SGPIO Waveform

Figure 29-2. Serial Data transmitted over the SGPIO Interface





30 System Management Interface and SMLink

30.1 Acronyms

Acronyms	Description
BMC	Baseboard Management Controller
EC	Embedded Controller

30.2 References

None

30.3 Overview

The PCH provides two SMLink interfaces, SMLink0 and SMLink1. The interfaces are intended for system management and are controlled by the Intel® CSME. Refer to the System Management chapter for more detail.

30.4 Signal Description

Name	Type	Description
INTRUDER#	I	Intruder Detect: This signal can be set to disable the system if box detected open.
SML0DATA/ GPP_C4	I/OD	System Management Link 0 Data: SMBus link to external PHY. External pull-up resistor is required.
SML0CLK / GPP_C3	I/OD	System Management Link 0 Clock: External pull-up resistor is required.
SML0ALERT# / GPP_C5	I/OD	System Management 0 Alert: Alert for the SMBus controller to optional Embedded Controller or BMC. External pull-up resistor is required.
SML1CLK / GPP_C6	I/OD	System Management Link 1 Clock: SMBus link to optional Embedded Controller or BMC. External pull-up resistor is required.
SML1DATA/ GPP_C7	I/OD	System Management Link 1 Data: SMBus link to optional Embedded Controller or BMC. External pull-up resistor is required.
SML1ALERT#/ PCHHOT# /GPP_B23	I/OD	System Management 1 Alert: Alert for the SMBus controller to optional Embedded Controller or BMC. A soft-strap determines the native function SML1ALERT# or PCHHOT# usage. External pull-up resistor is required.
PCHHOT# / GPP_B23 / SML1ALERT#	OD	PCHHOT#: This signal is used to indicate a PCH temperature out of bounds condition to an external EC. External pull-up resistor is required.
SML4DATA/ GPP_H17	I/OD	System Management Link 4 Data: External pull-up resistor is required.



30.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SML[2:0]ALERT#	Pull-down	20 kohm \pm 30%	
PCHHOT#	Pull-down	20 kohm \pm 30%	

30.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ¹	Immediately after Reset ¹	S3/S4/S5	Deep Sx
INTRUDER#	RTC	Undriven	Undriven	Undriven	Undriven
SML[4:0]DATA	Primary	Undriven	Undriven	Undriven	Off
SML[4:0]CLK	Primary	Undriven	Undriven	Undriven	Off
SML[2:0]ALERT#	Primary	Pull-down (Internal)	Driven Low	Driven Low	Off
SML[4:3]ALERT#	Primary	Undriven	Undriven	Undriven	Off
PCHHOT#	Primary	Pull-down (Internal)	Driven Low	Pull-down (Internal)	Off

Notes:
 1. Reset reference for primary well pins is RSMRST# and RTC well pins is RTCRST#.

30.7 Functional Description

The SMLink interfaces are controlled by the Intel[®] CSME.

Server Only: Note that there are 5 SMLink interfaces in server SKUs. SMLink0 is mainly used for integrated LAN. When an Intel LAN PHY is connected to SMLink0, a soft strap must be set to indicate that the PHY is connected to SMLink0. The interface will be running at a frequency of up to 1 MHz depending on different factors such as board routing or bus loading when the Fast Mode is enabled using a soft strap.

SMLink1 can be used with an Embedded Controller (EC) or Baseboard Management Controller (BMC).

Both SMLink0 and SMLink1 support up to 1 MHz.





31 Host System Management Bus (SMBus) Controller

31.1 Acronyms

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

31.2 References

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

31.3 Overview

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 I²C compatible devices.

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

31.4 Signal Description

Name	Type	Description
SMBCLK/ GPP_C0	I/OD	SMBus Clock. External Pull-up resistor is required.
SMBDATA/ GPP_C1	I/OD	SMBus Data. External Pull-up resistor is required.
SMBALERT#/ GPP_C2	I/OD	SMBus Alert: This signal is used to wake the system or generate SMI#. External Pull-up resistor is required.

31.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SMBALERT#	Pull-down	20 kohm \pm 30%	The integrated pull down is disabled after PCH_PWROK assertion



31.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ¹	Immediately after Reset ¹	S3/S4/S5	Deep Sx
SMBDATA	Primary	Undriven	Undriven	Undriven	Off
SMBCLK	Primary	Undriven	Undriven	Undriven	Off
SMBALERT#	Primary	Pull-down (Internal)	Driven Low	Driven Low	Off
Notes:					
1. Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.					

31.7 Functional Description

The PCH provides a 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 I²C 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.

31.7.1 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.

31.7.1.1 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.

31.7.1.2 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 I²C 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 E²PROMs. 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 I²C 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 [Table 31-1](#).

Table 31-1. I²C Block Read (Sheet 1 of 2)

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
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

**Table 31-1. I²C Block Read (Sheet 2 of 2)**

Bit	Description
-	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:

- $M \geq 1$ byte
- $N \geq 1$ byte
- $M + N \leq 32$ 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.

31.7.1.3 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.



31.7.1.4 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.

31.7.1.5 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).

31.7.1.6 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.

Table 31-2, Table 31-3 and Table 31-4 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 31-2. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	SMBALERT_DIS (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 31-3. Enables for 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 31-4. Enables 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)

31.7.1.7 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.

31.7.2 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

Note: The external microcontroller should not attempt to access the PCH SMBus slave logic until either:

- 800 milliseconds after both: RTCRST# is high and RSMRST# 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.

31.7.2.1 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.

Table 31-5 has the values associated with the registers.

Table 31-5. Slave Write Registers

Register	Function
0	Command Register. Refer Table 31-6 for valid values written to this register.
1-3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6-FFh	Reserved
Note: The external 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 31-6. Command Types (Sheet 1 of 2)

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 RSMRST# signal.



Table 31-6. Command Types (Sheet 2 of 2)

Command Type	Description
6	WD RELOAD: Reload watchdog timer.
7	Reserved
8	<p>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.</p> <p>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.</p>
9-FFh	Reserved.

31.7.2.2 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 31-7. 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 Table 31-8 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 Table 31-8 for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	



Table 31-8. Data Values for Slave Read Registers

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	7:0	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.
	2:1	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.
5	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.
	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 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



31.7.2.2.1 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.

31.7.2.3 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.

31.7.2.4 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.

Table 31-9 shows the Host Notify format.



Table 31-9. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Master	
2-8	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
11-17	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	
20-27	Data Byte Low - 8 bits	External Master	Loaded into the Notify Data Low Byte Register
28	ACK	PCH	
29-36	Data Byte High - 8 bits	External Master	Loaded into the Notify Data High Byte Register
37	ACK	PCH	
38	Stop	External Master	

31.7.2.5 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 31-10. 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 Table 31-11 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 Table 31-11 for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	



Table 31-11. Data Values for Slave Read Registers (Sheet 1 of 2)

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
2	7:0	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.
	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.
	4:3	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



Table 31-11. Data Values for Slave Read Registers (Sheet 2 of 2)

Register	Bits	Description
E	7:0	Month of the RTC
F	7:0	Year of the RTC
10h–FFh	7:0	Reserved

Table 31-12. Enables for 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

31.8 SMBus Power Gating

SMBus shares the Power Gating Domain with LPC and Primary-to-Sideband Bridge (P2SB).

A single FET controls the single Power Gating Domain; but LPC, SMBus and P2SB each has its own dedicated Power Gating Control Block.

The FET is only turned off when all these interfaces are ready to PG entry or already in the PG state.





32 Serial Peripheral Interface (SPI)

32.1 Acronyms

Acronyms	Description
CLK	Clock
CS	Chip Select
FCBA	Flash Component Base Address
FIBA	Flash Initialization Base Address
FLA	Flash Linear Address
FMBA	Flash Master Base Address
FPSBA	Flash PCH Strap Base Address
FRBA	Flash Region Base Address
MDTBA	MIP Descriptor Table Base Address
FRBA	Flash Region Base Address
MISO	Master In Slave Out
MOSI	Master Out Slave In
SFDP	Serial Flash Discovery Parameter
TPM	Trusted Platform Module

32.2 Overview

The PCH 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 SPI1 interface implementing 1 Chip Select signal (SPI1_CS#), is intended for integrated touch implementation. The SPI0 and SPI1 interfaces together support either 1.8 V or 3.3 V.

32.3 Signal Description

Table 32-1. Signal Description (Sheet 1 of 2)

Name	Type	Description
SPI0_CLK	0	SPI0 Clock: SPI clock signal for the common flash/TPM interface. Supports 20 MHz, 33 MHz and 50 MHz.
SPI0_CS0#	0	SPI0 Chip Select 0: Used to select the primary SPI0 Flash device. Note: This signal cannot be used for any other type of device than SPI Flash.
SPI0_CS1#	0	SPI0 Chip Select 1: Used to select an optional secondary SPI0 Flash device. Note: This signal cannot be used for any other type of device than SPI Flash.



Table 32-1. Signal Description (Sheet 2 of 2)

Name	Type	Description
SPIO_CS2#	O	SPIO Chip Select 2: Used to select the TPM device if it is connected to the SPIO interface. It cannot be used for any other type of device. Note: TPM can be configured through Soft Straps to operate over LPC or SPIO, but no more than 1 TPM is allowed in the system.
SPIO_MOSI	I/O	SPIO Master OUT Slave IN: Defaults as a data output pin for PCH in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPIO_IO0) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPIO_MISO	I/O	SPIO Master IN Slave OUT: Defaults as a data input pin for PCH in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPIO_IO1) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPIO_IO[3:2]	I/O	SPIO Data I/O: A bidirectional signal used to support Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes. This signal is not used in Dual Output Fast Read mode.
GPP_D1 / SPI1_CLK / BK1 / SBK1	O	SPI1 Clock: SPI1 Clock output from PCH
GPP_D2 / SPI1_MISO / BK2 / SBK2	I/O	SPI1 Master IN Slave OUT: SPI1 serial input data from the SPI1 Touch Screen device to PCH. This Pin will also function as Output during Dual and Quad I/O operation
GPP_D3 / SPI1_MOSI / BK3 / SBK3	I/O	SPI1 Master OUT Slave IN: SPI1 serial output data from PCH to the SPI1 Touch Screen device. This Pin will also function as Input during Dual and Quad I/O operation
GPP_D21 / SPI1_IO2	I/O	SPI1 Data I/O: SPI1 I/O to comprehend the support for the Quad I/O operation
GPP_D22 / SPI1_IO3	I/O	SPI1 Data I/O: SPI1 I/O to comprehend the support for the Quad I/O operation
GPP_D0 / SPI1_CS# / BK0 / SBK0	O	SPI1 Chip Select: SPI1 chip select

32.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SPIO_CLK	Pull-down	20 kohm ± 30%	
SPIO_MOSI	Pull-up	20 kohm ± 30%	Note
SPIO_MISO	Pull-up	20 kohm ± 30%	Note
SPIO_CS[2:0]#	Pull-down	20 kohm ± 30%	
SPIO_IO[2:3]	Pull-up	20 kohm ± 30%	Note

Note: The internal pull-up is disabled when RSMRST# is asserted (during reset) and only enabled after RSMRST# de-assertion

32.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset (Refer Note 1)	Immediately after Reset	S3/S4/S5	Deep Sx
SPIO_CLK	Primary	Internal Pull-down	Driven Low	Driven Low	OFF
SPIO_MOSI	Primary	Hi-Z (Refer Note 2)	Internal PU, then Driven Low	Driven Low	OFF



Signal Name	Power Plane	During Reset (Refer Note 1)	Immediately after Reset	S3/S4/S5	Deep Sx
SPI0_MISO	Primary	Hi-Z	Internal Pull-up	Internal Pull-up	OFF
SPI0_CS0#	Primary	Internal Pull-down	Driven High	Driven High	OFF
SPI0_CS1#	Primary	Internal Pull-down	Driven High	Driven High	OFF
SPI0_CS2#	Primary	Internal Pull-down	Driven High	Driven High	OFF
SPI0_IO[3:2]	Primary	Hi-Z (Refer Note 2)	Internal Pull-up	Internal Pull-up	OFF
Notes:					
1. During reset refers to when RSMRST# is asserted.					
2. SPI0_MOSI, SPI0_IO[3:2] also function as strap pins. The actual pin state during Reset is dependent on the platform Pull-up/Pull-down resistor.					

32.6 Functional Description

32.6.1 SPI0 for Flash

32.6.1.1 Overview

The Serial Peripheral Interface (SPI0) supports two SPI flash devices via 2 chip select (SPI0_CS0# and SPI0_CS1#). The maximum size of flash supported is determined by the SFDP-discovered addressing capability of each device. Each component can be up to 16 MB (32 MB total addressable) using 3-byte addressing. Each component can be up to 64 MB (128 MB total addressable) using 4-byte addressing. Another chip select (SPI0_CS2#) is also available and only used for TPM on SPI support. PCH drives the SPI0 interface clock at either 20 MHz, 33 MHz, or 50 MHz and will function with SPI flash devices that support at least one of these frequencies. The SPI interface supports either 3.3 V or 1.8 V.

A SPI0 flash device supporting SFDP (Serial Flash Discovery Parameter) is required for all PCH design. A SPI0 flash device on SPI0_CS0# 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)

The PCH SPI0 has a third chip select SPI0_CS2# for TPM support over SPI. The TPM on SPI0 will use SPI0_CLK, SPI0_MISO, SPI0_MOSI and SPI0_CS2# SPI signals.

Note: If Boot BIOS Strap = '1' then LPC is selected as the location for BIOS. BIOS may still be placed on LPC, but all platforms with PCH require a SPI0 flash connected to SPI bus with a valid descriptor connected to Chip Select 0 in order to boot.

32.6.1.2 SPI0 Supported Features

32.6.1.2.1 Descriptor Mode

Descriptor Mode is required for all SKUs of the PCH. Non-Descriptor Mode is not supported.



32.6.1.2.2 SPI0 Flash Regions

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

Table 32-2. SPI0 Flash Regions

Region	Content
0	Flash Descriptor
1	BIOS
2	Intel® Management Engine
3	Gigabit Ethernet
4	Platform Data
5	EC

Only four masters can access the regions: Host processor running BIOS code, Integrated Gigabit Ethernet and Host processor running Gigabit Ethernet Software, Intel® Management Engine, and the EC.

The Flash Descriptor and Intel® CSME region are the only required regions. The Flash Descriptor has to be in region 0 and region 0 must be located in the first sector of Device 0 (Offset 0). All other regions can be organized in any order.

Regions can extend across multiple components, but must be contiguous.

Flash Region Sizes

SPI0 flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4-KB or larger block. GbE requires two 4-KB or larger blocks. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the Intel® CSME and BIOS regions. The Intel® CSME region contains firmware to support Intel® Active Management Technology and other Intel® CSME capabilities.

Table 32-3. Region Size Versus Erase Granularity of Flash Components

Region	Size with 4-KB Blocks	Size with 8-KB Blocks	Size with 64-KB Blocks
Descriptor	4 KB	8 KB	64 KB
GbE	8 KB	16 KB	128 KB
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSME	Varies by Platform	Varies by Platform	Varies by Platform
EC	Varies by Platform	Varies by Platform	Varies by Platform

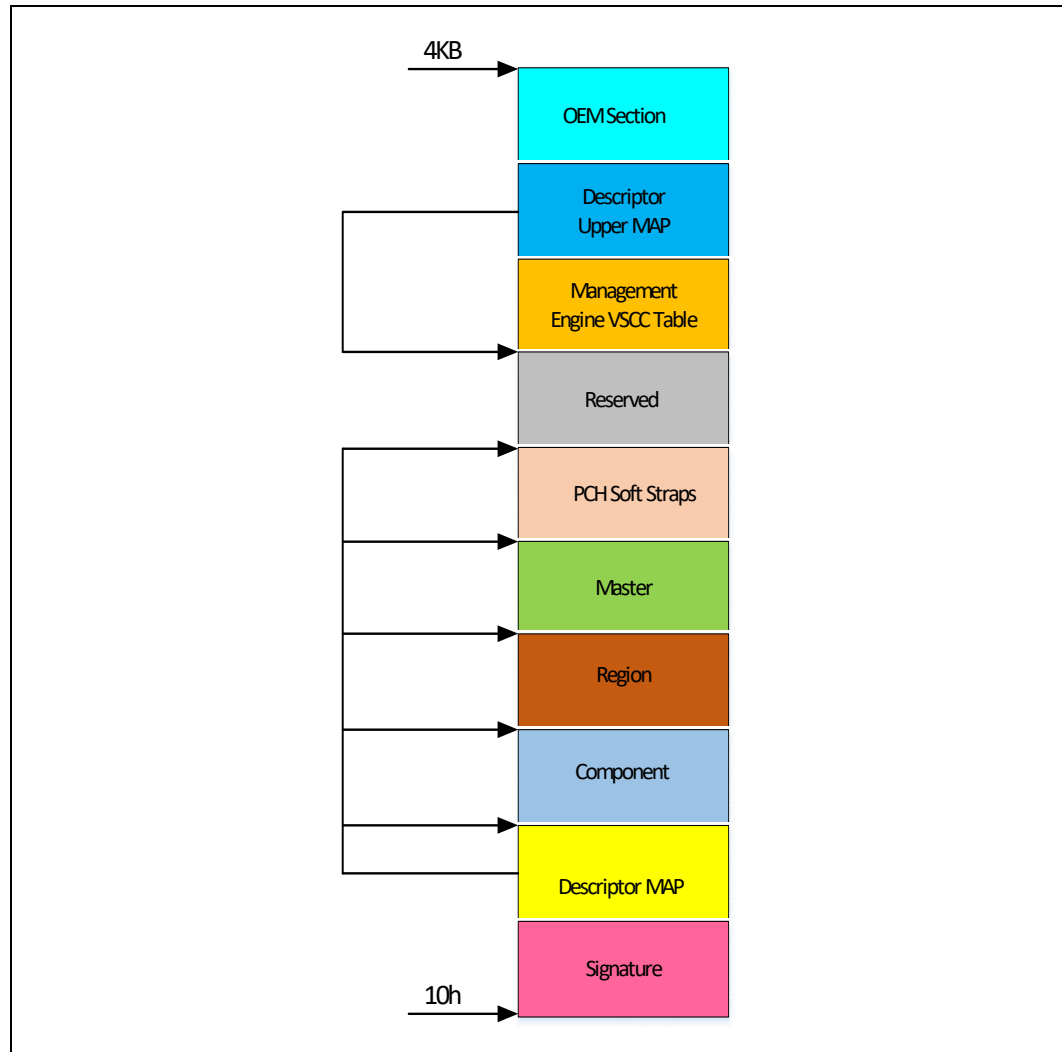
32.6.1.3 Flash Descriptor

The bottom sector of the flash component 0 contains the Flash Descriptor. The maximum size of the Flash Descriptor is 4 KB. If the block/sector size of the SPI0 flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. 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 region sections are shown in [Figure 32-1](#).



Figure 32-1. Flash Descriptor Regions



- The Flash 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 0FF0 A55Ah in order to be in Descriptor mode.
- The Descriptor map has pointers to the other five descriptor sections as well as the size of each.
- The component section has information about the SPI0 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.
- The Region section points to the three other regions as well as the size of each region.
- The master region contains the security settings for the flash, granting read/write permissions for each region and identifying each master by a requestor ID.
- The Reserved region between the top of the processor strap section and the bottom of the OEM Section is reserved for future chipset usages.



- The Descriptor Upper MAP determines the length and base address of the Management Engine VSCC Table.
- The Management Engine VSCC Table holds the JEDEC ID and the VSCC information of the entire SPI0 Flash supported by the NVM image.
- OEM Section is 256 bytes reserved at the top of the Flash Descriptor for use by OEM.

32.6.1.3.1 Descriptor Master Region

The master region defines read and write access setting for each region of the SPI0 device. The master region recognizes four masters: BIOS, Gigabit Ethernet, Management Engine, and EC. Each master is only allowed to do direct reads of its primary regions.

Table 32-4. Region Access Control Table

Master Access	Descriptor Region	BIOS Region	Intel® CSME	GbE Controller	EC
Intel® ME Read access	Yes	No	Yes	Yes	No
Intel® ME Write access	No	No	Yes	No	No
Gigabit Ethernet read access	Yes	No	No	Yes	No
Gigabit Ethernet write access	No	No	No	Yes	No
BIOS read access	Yes	Yes	No	Yes	Note
BIOS write access	No	Yes	No	Yes	Note
EC read access	Yes	Yes	No	No	Yes
EC write access	No	No	No	No	Yes

Note: Optional BIOS access to the EC region.

32.6.1.3.2 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® ME Straps	C3Ch
Register Init FIBA	340h

32.6.1.4 Flash Access

There are two types of accesses: Direct Access and Program Register Accesses.

32.6.1.4.1 Direct Access

- Masters are allowed to do direct read only of their primary region



- Gigabit Ethernet region can only be directly accessed by the Gigabit Ethernet controller. Gigabit Ethernet software must use Program Registers to access the Gigabit Ethernet region.
- Master's Host or Intel® Management Engine virtual read address is converted into the SPI0 Flash Linear Address (FLA) using the Flash Descriptor Region Base/Limit registers

Direct Access Security

- Requester ID of the device must match that of the primary Requester ID in the Master Section
- Calculated Flash Linear Address must fall between primary region base/limit
- Direct Write not allowed
- Direct Read Cache contents are reset to 0's on a read from a different master

32.6.1.4.2 Program Register Access

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

Register Access Security

- Only primary region masters can access the registers

32.6.2 SPI0 Support for TPM

The Serial Peripheral Interface (SPI0) supports a discrete TPM on the platform via its dedicated SPI0_CS2# signal.

SPI0 controller supports accesses to SPI0 TPM at approximately 17 MHz, 33 MHz and 48 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. SPI0 TPM device must support a clock of 20 MHz, and thus should handle 15-20 MHz. It may but is not required to support a frequency greater than 20 MHz.

TPM requires the support for the interrupt routing. However, the TPM's interrupt pin is routed to the PCH's PIRQ pin. Thus, TPM interrupt is completely independent from the SPI0 controller.

32.6.3 SPI1 Support for Touch Device

The Serial Peripheral Interface (SPI1) supports SPI1 touch device via chip select (SPI1_CS#) with Quad IO. The PCH drives the SPI1 interface clock at 30 MHz and will function with a SPI Touch device that supports this frequency.



33 Testability

33.1 JTAG

33.1.1 Acronyms

Acronyms	Description
OOB	Out of Band
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

33.1.2 References

Specification	Location
IEEE Standard Test Access Port and Boundary Scan Architecture	http://standards.ieee.org/findstds/standard/1149.1-2013.html

33.1.3 Overview

This section contains 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).

33.1.4 Signal Description

Table 33-1. Testability Signals (Sheet 1 of 2)

Name	Type	Description
PCH_JTAG_TCK	I/O	Test Clock Input (TCK): The test clock input provides the clock for the JTAG test logic.
PCH_JTAG_TMS	I/OD	Test Mode Select (TMS): The signal is decoded by the Test Access Port (TAP) controller to control test operations.
PCH_JTAG_TDI	I/OD	Test Data Input (TDI): Serial test instructions and data are received by the test logic at TDI.
PCH_JTAG_TDO	I/OD	Test Data Output (TDO): TDO is the serial output for test instructions and data from the test logic defined in this standard.
PCH_JTAGX	I/O	This pin is used to support merged debug port topologies.



Table 33-1. Testability Signals (Sheet 2 of 2)

Name	Type	Description
ITP_PMODE	I/O	ITP Power Mode Indicator. This signal is used to transmit processor and PCH power/reset information to the ITP Debugger.
PREQ#	I/O	Probemode Request: This signal is used to indicate a probe mode request send from a debugger to CPU.
PRDY#	I/O	Probemode Ready: This signal is used to acknowledge debugger that CPU in probe mode.
CPU_TRST#	I/O	CPU Test Reset: This active low signal is used to reset the test logic/Test Access Port (TAP) controller's state machine.

33.1.5 I/O Signal Planes and States

Table 33-2. Power Planes and States for Testability Signals

Signal Name	Power Plane	Resisters	During Reset ²	Immediately after Reset ²	S3/S4/S5	Deep Sx
PCH_JTAG_TCK	Primary	Strong Internal Pull-Down	L	L	L	OFF
PCH_JTAG_TMS	Primary	Internal Pull-Up	H	H	H	OFF
PCH_JTAG_TDI	Primary	Internal Pull-Up	H	H	H	OFF
PCH_JTAG_TDO	Primary	External Pull-Up	Z	Z	Z	OFF
PCH_JTAGX¹	Primary	Internal Strong Pull-Up (as TDO i/p), Internal Strong Pull-Down (as TCK o/p)	H	H/L	H/L	OFF
ITP_PMODE²	Primary	Internal Pull-Up	H	H	H	OFF
PREQ#	Primary	Internal Pull-Up	H	H	Z	OFF
PRDY#	Primary	Internal Pull-Up	H	H	Z	OFF
CPU_TRST#	Primary	Internal Pull-Down	L	L	L	OFF

Note:

- This signal is used in common JTAG topology to take in last device's TDO to DCI. The only planned supported topology is the Shared Topology. Thus, this pin will operate as TCK mode.
- Reset reference for primary well pins is RSMRST#

33.2 Intel® Trace Hub (Intel® TH)

33.2.1 Overview

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 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:

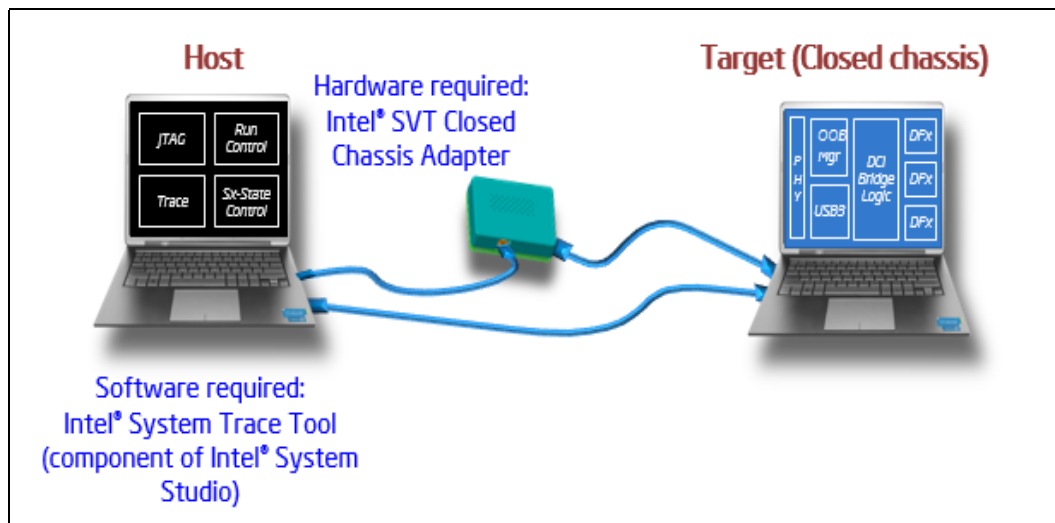
- Direct Connect Interface (DCI)
 - OOB Hosting DCI
 - USB 3.1 hosting DCI.DBC
- System Memory

There are multiple trace sources in the platform:

- BIOS
- Intel® CSME
- AET (Architecture Event Trace)
- Power Management Event Trace
- Hardware Signals
- Windows* ETW (for driver or application)

33.2.2 Platform Setup

Figure 33-1. Platform Setup with Intel® Trace Hub



33.3 Direct Connect Interface (DCI)

Direct Connect Interface (DCI) is a debug transport technology to enable closed chassis debug through any of USB 3.1 or USB 2.0 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 etc. 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 based debugger (kernel level debugger) are mutually exclusive.



There are two types of DCI hosting interfaces in the platform:

- OOB Hosting DCI
- USB 3.1 or USB 2.0 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

33.3.1 Out Of Band (OOB) Hosting DCI

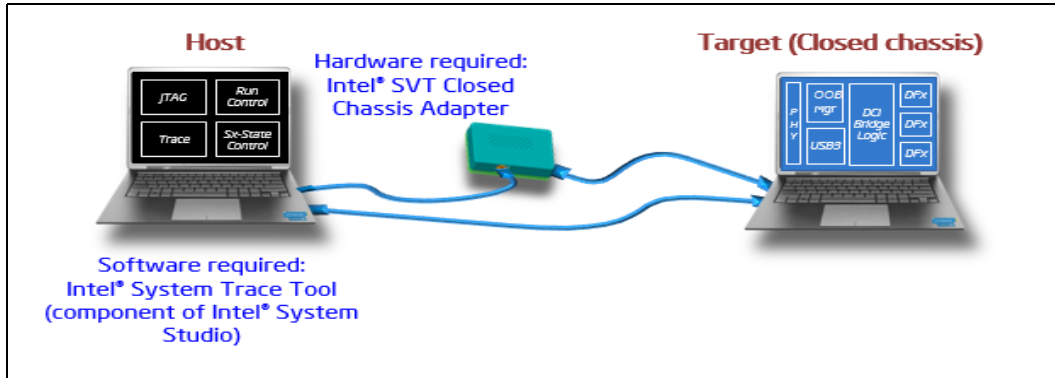
OOB was developed to provide an alternate path to convey controls and data to or from the embedded logic by connecting physically to the target through any USB 3.1 port. OOB provides an alternate side band path around the USB 3.1 Host controller, so that the embedded logic can be accessed, even when the USB controller is not alive (such as in low power states), or is malfunctioning. This path does not rely on USB 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 developed by Intel called Intel® SVT Closed Chassis Adapter (CCA). It is a simple data transformation device. This adapter generates a OOB signaling protocol operating at up to 100 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 the moral equivalent of the USB link layer. This path also uses native DCI packet protocol instead of USB 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.

33.3.2 USB 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 device for debug purpose. This path uses the USB packet protocol layer, USB link layer flow control and USB physical layer. USB 3.1 only works in S0. USB 2.0 survives Sx states and provides early boot access.

33.3.3 Platform Setup

Figure 33-2. Platform Setup with DCI Connection



§ §



34 Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers

34.1 Acronyms

Acronyms	Description
DMA	Direct Memory Access
UART	Universal Asynchronous Receiver/Transmitter

34.2 Overview

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

Note: Bluetooth* devices are not supported on the PCH UART interfaces.

34.3 Signal Description

Name	Type	Description
UART0A_RXD/ GPP_C8	I	UART 0 Receive Data
UART0A_TXD/ GPP_C9	O	UART 0 Transmit Data
UART0A_RTS#/ GPP_C10	O	UART 0 Request to Send
UART0A_CTS#/ GPP_C11	I	UART 0 Clear to Send
UART0B_RXD / GPP_J5 / CNV_BRI_RSP	I	Second Instant of UART 0 Receive Data
UART0B_TXD / GPP_J6 / CNV_RGI_DT	O	Second Instant of UART 0 Transmit Data
UART0B_RTS#/ GPP_J4 / CNV_BRI_DT	O	Second Instant of UART 0 Request to Send



Name	Type	Description
UART0B_CTS# / GPP_J7 / CNV_RGI_RSP	I	Second Instant of UART 0 Clear to Send
UART1_RXD / ISH_UART1_RXD/ GPP_C12	I	UART 1 Receive Data
UART1_TXD / ISH_UART1_TXD/ GPP_C13	O	UART 1 Transmit Data
UART1_RTS# / ISH_UART1_RTS#/ GPP_C14	O	UART 1 Request to Send
UART1_CTS# / ISH_UART1_CTS#/ GPP_C15	I	UART 1 Clear to Send
UART2_RXD / GPP_C20	I	UART 2 Receive Data
UART2_TXD / GPP_C21	O	UART 2 Transmit Data
UART2_RTS# / GPP_C22	O	UART 2 Request to Send
UART2_CTS# / GPP_C23	I	UART 2 Clear to Send

34.4 Integrated Pull-Ups and Pull-Downs

None

34.5 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#	Primary	Undriven	Undriven	Undriven	Off
UART[2:0]_CTS#	Primary	Undriven	Undriven	Undriven	Off
Notes:					
1. Reset reference for primary well pins is RSMRST#.					

34.6 Functional Description

34.6.1 Features

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: Note the following:

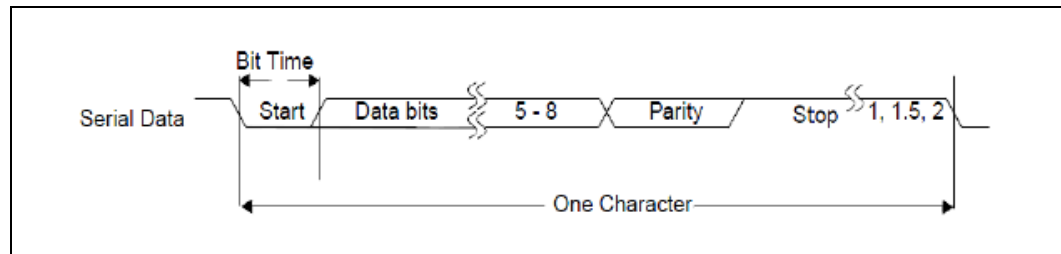
1. SIR mode is not supported.
2. External read enable signal for RAM wake up when using external RAMs is not supported.

34.6.2 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 34-1. UART Serial Protocol



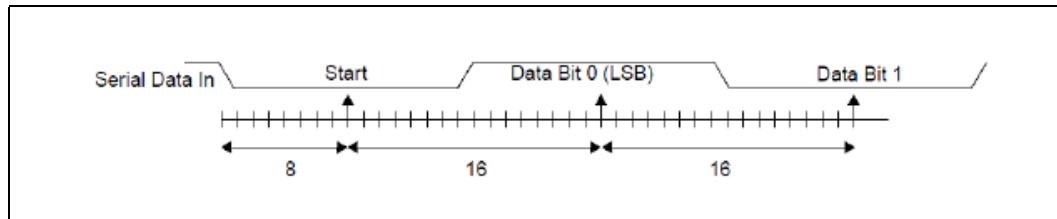
The UART Host Controller Line Control Register (LCR) is used to control the serial character characteristics. The individual bits of the data word are sent after the Start bit, starting with the least significant bit (LSB). These are followed by the optional parity bit, followed by the Stop bit(s), which 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 34-2. UART Receiver Serial Data Sample Points



34.6.3 16550 8-bit Addressing - Debug Driver Compatibility

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.

Note: The UART 16550 8-bit Legacy mode only operates with PIO transactions. DMA transactions are not supported in this mode.

34.6.4 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.

34.6.4.1 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.



34.6.4.2 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.

34.6.5 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.

34.6.6 Power Management

34.6.6.1 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.

34.6.6.2 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.

34.6.7 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.

34.6.8 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.

§ §



35 Universal Serial Bus (USB)

35.1 Acronyms

Acronyms	Description
xHCI	eXtensible Host Controller Interface

35.2 References

Specification	Location
USB 3.1 Specification	www.usb.org
USB 3.0 Specification	www.usb.org
USB 2.0 Specification	www.usb.org

35.3 Overview

The PCH implements an xHCI USB 3.1 controller which provides support for up to 14 USB 2.0 signal pairs and 10 USB 3.1 signal pairs. The xHCI controller supports wake up from sleep states S1-S4. The xHCI controller supports up to 64 devices for a max number of 2048 Asynchronous endpoints (Control / Bulk) or max number of 128 Periodic endpoints (Interrupt / isochronous).

Note: Each walk-up USB 3.1 capable port must have USB 3.1 signaling and USB 2.0 signaling.

35.4 Signal Description

Table 35-1. Signal Descriptions (Sheet 1 of 4)

Name	Type	Description
USB31_1_RXN, USB31_1_RXP	I	USB 3.1 Differential Receive Pair 1: These are USB 3.1-based high-speed differential signals for Port #1 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_1_TXN, USB31_1_TXP	O	USB 3.1 Differential Transmit Pair 1: These are USB 3.1-based high-speed differential signals for Port #1 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals
USB31_2_RXN, USB31_2_RXP	I	USB 3.1 Differential Receive Pair 2: These are USB 3.1-based high-speed differential signals for Port #2 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals
USB31_2_TXN, USB31_2_TXP	O	USB 3.1 Differential Transmit Pair 2: These are USB 3.1-based high-speed differential signals for Port #2 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals
USB31_3_RXN, USB31_3_RXP	I	USB 3.1 Differential Receive Pair 3: These are USB 3.1-based high-speed differential signals for Port #3 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_3_TXN, USB31_3_TXP	O	USB 3.1 Differential Transmit Pair 3: These are USB 3.1-based high-speed differential signals for Port #3 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.



Table 35-1. Signal Descriptions (Sheet 2 of 4)

Name	Type	Description
USB31_4_RXN, USB31_4_RXP	I	USB 3.1 Differential Receive Pair 4: These are USB 3.1-based high-speed differential signals for Port #4 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals
USB31_4_TXN, USB31_4_TXP	O	USB 3.1 Differential Transmit Pair 4: These are USB 3.1-based high-speed differential signals for Port #4 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_5_RXN, USB31_5_RXP	I	USB 3.1 Differential Receive Pair 5: These are USB 3.1-based high-speed differential signals for Port #5 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_5_TXN, USB31_5_TXP	O	USB 3.1 Differential Transmit Pair 5: These are USB 3.1-based high-speed differential signals for Port #5 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_6_RXN, USB31_6_RXP	I	USB 3.1 Differential Receive Pair 6: These are USB 3.1-based high-speed differential signals for Port #6 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_6_TXN, USB31_6_TXP	O	USB 3.1 Differential Transmit Pair 6: These are USB 3.1-based high-speed differential signals for Port #6 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_7_RXN / PCIE1_RXN, USB31_7_RXP / PCIE1_RXP	I	USB 3.1 Differential Receive Pair 7: These are USB 3.1-based high-speed differential signals for Port #7 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_7_TXN / PCIE1_TXN, USB31_7_TXP / PCIE1_TXP	O	USB 3.1 Differential Transmit Pair 7: These are USB 3.1-based high-speed differential signals for Port #7 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_8_RXN / PCIE2_RXN, USB31_8_RXP / PCIE2_RXP	I	USB 3.1 Differential Receive Pair 8: These are USB 3.1-based high-speed differential signals for Port #8 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_8_TXN / PCIE2_TXN, USB31_8_TXP / PCIE2_TXP	O	USB 3.1 Differential Transmit Pair 8: These are USB 3.1-based high-speed differential signals for Port #8 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_9_RXN / PCIE3_RXN, USB31_9_RXP / PCIE3_RXP	I	USB 3.1 Differential Receive Pair 9: These are USB 3.1-based high-speed differential signals for Port #9 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_9_TXN / PCIE3_TXN, USB31_9_TXP / PCIE3_TXP	O	USB 3.1 Differential Transmit Pair 9: These are USB 3.1-based high-speed differential signals for Port #9 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_10_RXN / PCIE4_RXN, USB31_10_RXP / PCIE4_RXP	I	USB 3.1 Differential Receive Pair 10: These are USB 3.1-based high-speed differential signals for Port #10 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB31_10_TXN / PCIE4_TXN, USB31_10_TXP / PCIE4_TXP	O	USB 3.1 Differential Transmit Pair 10: These are USB 3.1-based high-speed differential signals for Port #10 and the xHCI Controller. This signal should be mapped to a USB connector and with one of the OC (overcurrent) signals.
USB2P_1, USB2N_1	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 1: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_2, USB2N_2	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 2: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OCOC (overcurrent) signals.



Table 35-1. Signal Descriptions (Sheet 3 of 4)

Name	Type	Description
USB2P_3, USB2N_3	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 3: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_4, USB2N_4	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 4: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_5, USB2N_5	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 5: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_6, USB2N_6	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 6: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_7, USB2N_7	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 7: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_8, USB2N_8	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 8: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_9, USB2N_9	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 9: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_10, USB2N_10	I/O	USB 2.0 Port 1 Transmit/Receive Differential Pair 10: This USB 2.0 signal pair are routed to xHCI Controller and should be mapped to a USB connector with one of the overcurrent OC (overcurrent) signals.
USB2P_11, USB2N_11	I/O	USB 2.0 Port 11 Transmit/Receive Differential Pair 11: This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector and with one of the OC (overcurrent) signals.
USB2P_12, USB2N_12	I/O	USB 2.0 Port 12 Transmit/Receive Differential Pair 12: This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector and with one of the OC (overcurrent) signals.
USB2P_13, USB2N_13	I/O	USB 2.0 Port 13 Transmit/Receive Differential Pair 13: This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector and with one of the OC (overcurrent) signals.
USB2P_14, USB2N_14	I/O	USB 2.0 Port 14 Transmit/Receive Differential Pair 14: This USB 2.0 This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector and with one of the OC (overcurrent) signals.
USB2_OC0#/ GPP_E9	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC1# / GPP_E10	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC2#/ GPP_E11	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC3#/ GPP_E12	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC4# / GPP_F15	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC5# / GPP_F16	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC6# / GPP_F17	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_OC7# / GPP_F18	I	Overcurrent Indicators: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.



Table 35-1. Signal Descriptions (Sheet 4 of 4)

Name	Type	Description
USB2_VBUSSENSE	I	VBUS Sense for USB Device mode. Refer to Device mode specification for the sensing threshold voltage spec. Note: This HW signal is not used on the PCH for USB device mode functionality. This signal should be connected to ground.
USB2_ID	I	ID detect for USB Device mode. Note: This HW signal is not used on the PCH for dual role mode selection. (OOB) LPC SCI/SMI
USB2_COMP	I	USB Resistor Bias, analog connection points for an external resistor to ground.

Note: It is not recommended to route the USB 3.1 signals to the USB connector for USB ports that is USB 2.0 capable only.

35.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
USB2N_[14:1]	Internal Pull-down	14.25–24.8 kohm	1
USB2P_[14:1]	Internal Pull-down	14.25–24.8 kohm	1
USB2_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%)

35.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset ²	Immediately after Reset ²	S3/S4/S5	Deep Sx
USB31_[10:1]_RXN USB31_[10:1]_RXP	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB31_[10:1]_TXN USB31_[10:1]_TXP	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB2N_[14:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB2P_[14:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB2_OC0#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC1#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC2#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC3#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC4#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC5#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC6#	Primary	Undriven	Undriven	Undriven	OFF
USB2_OC7#	Primary	Undriven	Undriven	Undriven	OFF
USB2_VBUSSENSE	Primary	Undriven	Undriven	Undriven	OFF



Signal Name	Power Plane	During Reset ²	Immediately after Reset ²	S3/S4/S5	Deep Sx
USB2_ID ¹	Primary	Internal Pull-UP	Undriven/ Internal Pull-UP	Undriven/ Internal Pull-UP	OFF
USB2_COMP	Primary	Undriven	Undriven	Undriven	OFF
Note: 1. The USB2_ID pin is pulled up internally. 2. Reset reference for primary well pins is RSMRST#, DSW well pins is DSW_PWROK, and RTC well pins is RTCRST#.					

35.7 Functional Description

35.7.1 eXtensible Host Controller Interface (xHCI) Controller (D20:F0)

The PCH contains an eXtensible Host Controller Interface (xHCI) which supports up to 14 USB 2.0 ports and up to 10 USB 3.1 ports with board routing, ACPI table and BIOS considerations are required. This controller allows data transfers of up to 10 Gb/s for USB 3.1 Gen 2 ports and 5 Gb/s for USB 3.1 Gen 1 ports. The controller supports SuperSpeedPlus (SS+), SuperSpeed (SS), High-Speed (HS), Full-Speed (FS) and Low-Speed (LS) traffic on the bus. The xHCI supports USB Debug port on all USB 3.1 capable ports. The xHCI also supports USB Attached SCIS Protocol (UASP). The PCH also support dual role capability.

35.7.2 USB Dual Role Support - eXtensible Device Controller Interface (xHCI) Controller (D20:F1)

The PCH also supports Dual Role Capability. The xHCI can now be paired with a standalone eXtensible Device Controller Interface (xDCI) to provide dual role functionality. The USB subsystem incorporates a xDCI USB 3.0 device controller (5Gb/s). The Device controller support SuperSpeed (SS), High-Speed (HS). This controller is instantiated as a separate PCI function. The PCH USB implementation is compliant to the Device specification and supports host/device through USB Type-C* connector.

The xDCI shares all USB ports with the host controller, with ownership of the port being decided based the USB Power Delivery specification. Since all the ports support device mode, xDCI enabling must be extended by System BIOS and EC. While the port is mapped to the device controller, the host controller Rx detection must always indicate a disconnected port.

35.7.3 Supported USB 2.0 Ports

Due to the USB 2.0 port requirement for integrated Bluetooth functionality with the integrated Intel® Wireless-AC (CNVi) solution, PCH-H USB 2.0 port 14 will be available.

If integrated Bluetooth functionality is not desired, customers may utilize the port for USB functionality.



36 Connectivity Integrated (CNVi)

36.1 Acronyms

Acronyms	Description
CNVi	Connectivity Integrated
IP	Literally, Intellectual Property. IP refers to architecture, design, validation, and software components collectively delivered to enable one or more specific SoC features
MFUART	Multifunction Universal Asynchronous Receiver/Transmitter
RGI	Radio Generic interface
SoC	System On Chip
UART	Universal Asynchronous Receiver/Transmitter

36.2 References

Specification	Location
M.2 Specification	https://pcisig.com/specifications/pciexpress/M.2_Specification/
MIP1 [®] Alliance specification for D-PHY v1.1	https://mipi.org/specifications

36.3 Overview

Connectivity Integrated (CNVi) is a general term referring to a family of connectivity solutions which are based on the Connectivity Controller family. The common component of all these solutions is the Connectivity Controller IP, which is a hard macro embedded in various Intel SoC chips.

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

- The containing chip (SoC or PCH which contains the Connectivity Controller IP)
- Buttress (as applicable to each platform)
- Companion RF chip that is in a pre-certified module (that is, M.2) or soldered as chip on board.

36.4 Signal Description

Name	Type	Description
GPIO Fixed Function		
GPP_D5 / I2S2_SFRM / CNV_RF_RESET#	I/O	For CNVi: RF companion (CRF) reset signal, active low, must be low and glitch free at power up. For standard CNV with UART host support: Optional Bluetooth* I2S bus sync.
GPP_D6 / I2S2_TXD / MODEM_CLKREQ	O	For CNVi: Clock request signal. Used to request the RF companion clock (38.4M Ref clock) for Pulsar and the PCH. For standard CNV with UART host Bluetooth* support: Optional Bluetooth* I ² S bus data output (input to Bluetooth* module).



Name	Type	Description
GPP_D7 / I2S2_RXD	I	For CNVi: Unused. For standard CNV with UART host support: Optional Bluetooth* I ² S bus data output (input to Bluetooth* module).
GPP_D8 / I2S2_SCLK	I	For CNVi: Unused. For standard CNV with UART host support: Optional Bluetooth* I ² S bus clock (input to Bluetooth* module).
GPP_J0 / CNV_PA_BLANKING	I	For CNVi and standard CNV: Optional WLAN/Bluetooth*-WWAN coexistence signal COEX3. Co-existence signal for external GNSS solution
GPP_J8 / CNV_MFUART2_RXD	I	For CNVi and standard CNV: Optional WLAN/Bluetooth*-WWAN coexistence signal COEX2.
GPP_J9 / CNV_MFUART2_TXD	O	For CNVi and standard CNV: Optional WLAN/Bluetooth*-WWAN coexistence signal COEX1.
GPP_J4 / CNV_BRI_DT	O	For CNVi: BRI bus TX. For standard CNV CNV with UART host support: Bluetooth* UART RTS#
GPP_J5 / CNV_BRI_RSP	I	For CNVi: BRI bus RX. For standard CNV CNV with UART host support: Bluetooth* UART RXD
GPP_J6 / CNV_RGI_DT	O	For CNVi: RGI bus TX. For standard CNV with UART host support: Bluetooth* UART TXD
GPP_J7 / CNV_RGI_RSP	I	For CNVi: RGI bus RX. For standard CNV with UART host support: Bluetooth* UART CTS#
GPIO Selectable Function		
GPP_C2	O	UART_BT_WAKE output. Optional to connect to Bluetooth* Wake on Bluetooth* Module. This is the recommended GPIO but other GPIOs can be selected for this function.
GPP_B3	O	BT_KILL output. Optional to connect to a Bluetooth* KILL pin on the Bluetooth* module. This is the recommended GPIO but other GPIOs can be selected for this function.
GPP_C5	O	For CNVi: Unused Wi-Fi* PCIe host wake output for standard CNV. Optional to connect to a WLAN PCIe host wake pin on the Wi-Fi* module. This is the recommended GPIO but other GPIOs can be selected for this function.
GPP_B4	O	WLAN_KILL output. Optional to connect to a WLAN KILL pin on the Wi-Fi* module. This is the recommended GPIO but other GPIOs can be selected for this function.
GPP_B8	O	For CNVi: Unused Wi-Fi* PCIe clock request output for standard CNV. Optional to connect to a WLAN PCIe clock request pin on the Wi-Fi* module. This is the recommended GPIO but other GPIOs can be selected for this function
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+



Name	Type	Description
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-
CLKIN_XTAL	I	Single-ended clock signal input for CNVi (38.4 MHz). Should be connected to the reference clock output of the RF companion (CRF)
CNV_WT_RCOMP	I	An analog pin, direct bump for external compensation resistor (RCOMP) for CNVio Interface. Connected to supply VCCDPHY_1P24
Selectable Special Purpose I/O		
USB2P_P14		Bluetooth* USB host bus (positive) for standard CNV. Optional to connect to a Bluetooth* USB+ pin on the Bluetooth* module. Port 14 is the recommended port but other USB2 ports can be selected for this function.
USB2N_P14		Bluetooth* USB host bus (negative) for standard CNV. Optional to connect to a Bluetooth* USB+ pin on the Bluetooth* module. Port 14 is the recommended port but other USB2 ports can be selected for this function.
PCIE8_TXP	O	Wi-Fi* PCIe host bus TX (positive) for standard CNV. Optional to connect to a Wi-Fi* PCIe PERp0 pin on the Wi-Fi* module. This is the recommended port but other PCIe ports can be selected for this function.
PCIE8_TXN	O	Wi-Fi* PCIe host bus TX (negative) for standard CNV. Optional to connect to a Wi-Fi* PCIe PERn0 pin on the Wi-Fi* module. This is the recommended port but other PCIe ports can be selected for this function.
PCIE8_RXP	I	Wi-Fi* PCIe host bus RX (positive) for standard CNV. Optional to connect to a Wi-Fi* PCIe PETp0 pin on the Wi-Fi* module. This is the recommended port but other PCIe ports can be selected for this function.
PCIE8_RXN	I	Wi-Fi* PCIe host bus RX (negative) for standard CNV. Optional to connect to a Wi-Fi* PCIe PETn0 pin on the Wi-Fi* module. This is the recommended port but other PCIe ports can be selected for this function.
CLKOUT_PCIE_P3	O	Wi-Fi* PCIe host bus clock (positive) for standard CNV. Optional to connect to a Wi-Fi* PCIe REFCLKp pin on the Wi-Fi* module. This is the recommended clock signal but other PCIe clocks can be selected for this function.
CLKOUT_PCIE_N3	O	Wi-Fi* PCIe host bus clock (negative) for standard CNV. Optional to connect to a Wi-Fi* PCIe REFCLKp pin on the Wi-Fi* module. This is the recommended clock signal but other PCIe clocks can be selected for this function.
CL_RST#	O	Wi-Fi* Controller Link host bus reset for standard CNV with Controller Link support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK reset pin on the Intel® vPro™ Wi-Fi* module.
CL_DATA	I/O	Wi-Fi* Controller Link host bus data for standard CNV with Controller Link support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK data pin on the Intel® vPro™ Wi-Fi* module.
CL_CLK	O	Wi-Fi* Controller Link host bus clock for standard CNV with Controller Link support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK clock pin on the Intel® vPro™ Wi-Fi* module.

36.5 I/O Signal Planes and States

Signal Name	Power plane	During Reset*	Immediately After Reset*	S3/S4/S5	Deep Sx
I2S2_RXD	Primary	Undriven	Undriven	Undriven	OFF
I2S2_SCLK	Primary	Undriven	Undriven	Undriven	OFF



Signal Name	Power plane	During Reset*	Immediately After Reset*	S3/S4/S5	Deep Sx
I2S2_TXD	Primary	Undriven	Undriven	Undriven	OFF
I2S2_SFRM	Primary	Undriven	Undriven	Undriven	OFF
CNV_RF_RESET#	Primary	Driven	Driven	Driven	OFF
MODEM_CLKREQ	Primary	Driven	Driven	Driven	OFF
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)	Powered (input)	Powered (input)	OFF
CNV_RGI_DT	Primary	Driven	Driven	Driven	OFF
CNV_RGI_RSP	Primary	Powered (input)	Powered (input)	Powered (input)	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
CLKIN_XTAL_LCP	Primary	Depends on MODEM_CLKREQ	Depends on MODEM_CLKREQ	Depends on MODEM_CLKREQ	OFF

Notes: *Reset reference for primary well pins is RSMRST#.



36.6 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.





37 GPIO Serial Expander

37.1 Acronyms, Definitions

Acronyms	Description
GSX	GPIO Serial Expander

37.2 Overview

GPIO Serial Expander (GSX) is the capability provided by the PCH to expand the GPIOs on a platform that needs more GPIOs than the ones provided by the PCH. The solution requires external shift register discrete components.

37.3 Signal Description

Name	Type	Description
GSXDOUT / GPP_K12	O	GPIO Serial Expander Controller Data Out
GSXSLOAD / GPP_K13	O	GPIO Serial Expander Controller Serial Load
GSXDIN / GPP_K14	I	GPIO Serial Expander Controller Data In
GSXSRESET# / GPP_K15	O	GPIO Serial Expander Controller Serial Reset
GSXCLK / GPP_K16	O	GPIO Serial Expander Controller Clock

37.4 Integrated Pull-ups and Pull-downs

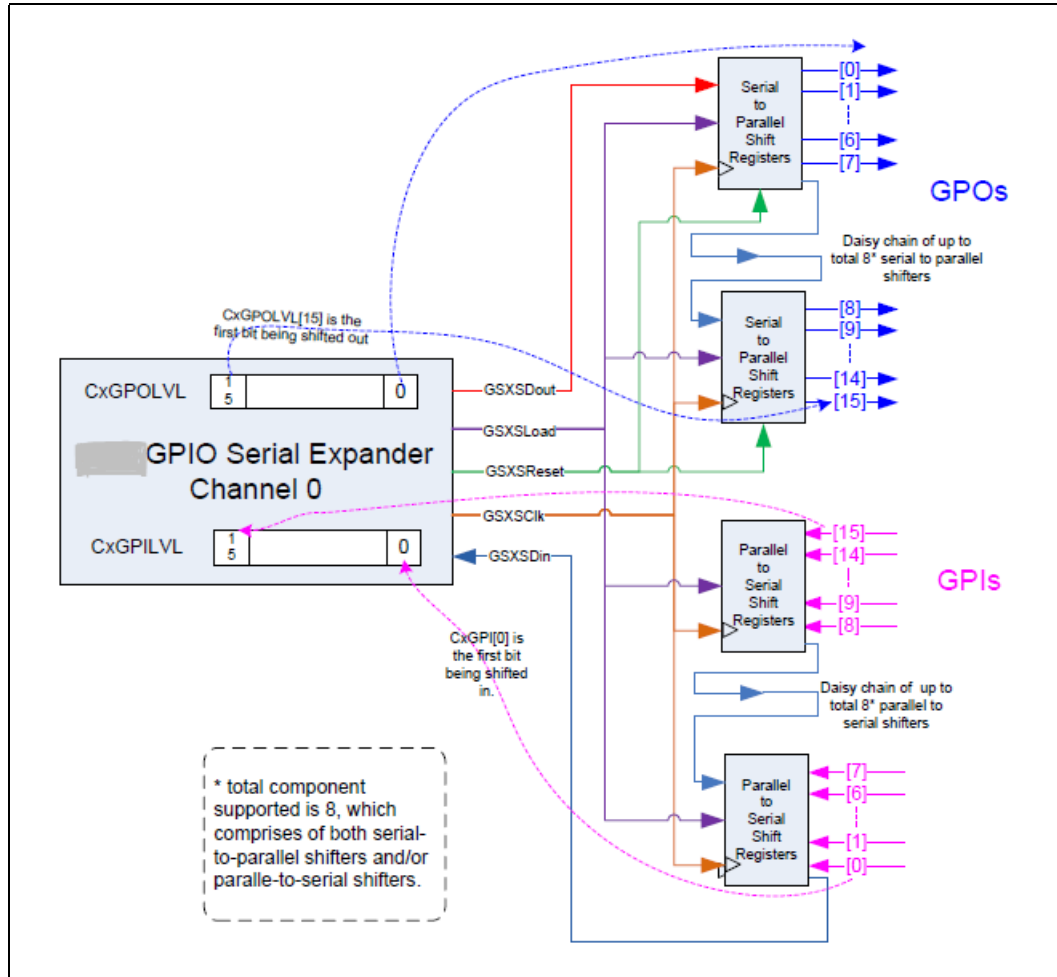
None

37.5 Functional Description

GPIO Serial Expander (GSX) uses serial-to-parallel or parallel-to-serial shift register discrete components to increase number of the GPIO pins for system use. It expands in the multiples of 8 for input or output with 8 pins per expander. The total shift register component supported is 8, which can expand the GPIOs by up to 64.

Figure 37-1 illustrates a GPIO expansion topology with 16 GPIs and 16 GPOs.

Figure 37-1. Example of GSX Topology



Coming out of system reset, GSX is in reset with the following behaviors:

- GSXSRESET# asserted by default. The signal remains asserted until BIOS/SW initialization has been completed and CxCMD.ST set to 1.
- GSXSLOAD is 0 by default until CxCMD.ST is set to 1.
- GSXSCLK is not toggling until CxCMD.ST is set to 1.

§ §



38 Direct Media Interface

38.1 Acronyms

Acronyms	Description
DMI	Direct Media Interface

38.2 Overview

The PCH communicates with the processor using high speed DMI that supports 8 GT/s, 5 GT/s and 2.5 GT/s data rates.

38.3 Signal Description

Name	Type	Description
DMI_RXN[3:0], DMI_RXP[3:0]	I	DMI receive lanes
DMI_TXN[3:0], DMI_TXP[3:0]	O	DMI transmit lanes

38.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset*	Immediately after Reset*	S3/S4/S5	Deep Sx
DMI_RXN[3:0] DMI_RXP[3:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
DMI_TXN[3:0] DMI_TXP[3:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off

Note: *Reset reference for primary well pins is RSMRST#.

38.5 Functional Description

The DMI Link supports a maximum of x4 widths

38.5.1 Lane Reversal

x4 DMI PCH Lane Reversal is supported. OEM can enable this feature in PCH to ease the DMI Link routing (between CPU and PCH) on the board. PCH DMI Lane Reversal is enabled or disabled through a SoftStrap.

38.5.2 Polarity Inversion

Polarity Inversion is supported on all the Receiver Lanes in PCH DMI. PCH DMI will autonomously detects the polarity inversion (Rx+ and Rx- is connected reversed) based on the Training Sequence received and enabled it during Link Training.





39 Private Configuration Space Target Port ID

39.1 Overview

The PCH incorporates a wide variety of devices and functions. The registers within these devices are mainly accessed through the primary interface, such as PCI configuration space and IO/MMIO space. Some devices also have registers that are distributed within the PCH Private Configuration Space at individual endpoints which are only accessible through the PCH Sideband Interface. Refer to Volume 2 for registers that reside in the PCH Private Configuration Space (PCR registers).

These PCH Private Configuration Space Registers can be addressed via SBREG_BAR, Target Port ID, and register offset.

The following table lists the PCH Target Port IDs for PCR register access.

Table 39-1. Private Configuration Space Register Target Port IDs (Sheet 1 of 2)

PCH Device/Function Type	Target Port ID
General Purpose I/O (GPIO) Community 4	6Ah
General Purpose I/O (GPIO) Community 3	6Bh
General Purpose I/O (GPIO) Community 2	6Ch
General Purpose I/O (GPIO) Community 1	6Dh
General Purpose I/O (GPIO) Community 0	6Eh
DCI	B8h
PSF1	BAh
PSF2	BBh
PSF3	BCh
ISH Controller	BEh
Real Time Clock (RTC)	C3h
Processor Interface, 8254 Timer, HPET, APIC	C4h
SMBus	C6h
LPC	C7h
USB 2.0	CAh
UART, I ² C, GSPI	CBh
FIA Configuration	CFh
SATA	D9h
Integrated Clock Controller (ICC)	DCh
PCIe* Controller #1 (SPA)	80h
PCIe* Controller #2 (SPB)	81h
PCIe* Controller #3 (SPC)	82h
PCIe* Controller #4 (SPD)	83h



Table 39-1. Private Configuration Space Register Target Port IDs (Sheet 2 of 2)

PCH Device/Function Type	Target Port ID
PCIe* Controller #5 (SPE)	84h
PCIe* Controller #6 (SPF)	85h
USB Dual Role / OTG	E5h
eSPI / SPI	72h
DMI Configuration	88h
Intel® Trace Hub	B6h
CNVi	73h

§ §



40 Secure Digital eXtended Capacity (SDXC)

40.1 Acronyms

Acronyms	Description
SDXC	Secure Digital eXtended Capacity

40.2 References

Specification	Location
SDXC Specifications	http://www.sdcard.org

40.3 Overview

The SDXC controller is to connect to an external detachable storage devices. It supports SDXC specification version 3.01.

40.4 Signal Description

Name	Type	Description	Voltage
SD_CMD /GPP_G0	I/O	SD Command/Response	3.3V or 1.8V
SD_DATA0 /GPP_G1	I/O	SD Data	3.3V or 1.8V
SD_DATA1 /GPP_G2	I/O	SD Data	3.3V or 1.8V
SD_DATA2 /GPP_G3	I/O	SD Data	3.3V or 1.8V
SD_DATA3 /GPP_G4	I/O	SD Data	3.3V or 1.8V
SD_CD# /GPP_G5	I	SDXC Detect	3.3V or 1.8V
SD_CLK /GPP_G6	O	SD clock	3.3V or 1.8V
SD_WP /GPP_G7	I	SDXC write protect	3.3V or 1.8V
SD_VDD1_PWR_EN# / ISH_GP7 /GPP_A17	O	SDXC power enable for 3.3V Note: Signal assertion logic can be changed by BIOS. (Active High is default assertion logic)	1.8V
SD_3P3_RCOMP	I/O	Impedance compensation for 3.3V operation of SD card and GPIO buffers. External reference resistor (200 Ohm +/- 1% pull down to ground) is required, regardless of SD card interface is used or not.	NA
SD_1P8_RCOMP	I/O	Impedance compensation for 1.8V operation of SD card and GPIO buffers. External reference resistor (200 Ohm +/- 1% pull down to ground) is required, regardless of SD card interface is used or not. The RCOMP resistor can be shared with SD_3P3_RCOMP.	NA



40.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset (Note)	Immediately after Reset (Note)	S3/S4/S5	Deep Sx
SD_CMD	Primary	Undriven	Undriven	Undriven	OFF
SD_DATA[3:0]	Primary	Undriven	Undriven	Undriven	OFF
SD_CD#	Primary	Undriven	Undriven	Undriven	OFF
SD_CLK	Primary	Undriven	Undriven	Undriven	OFF
SD_WP	Primary	Undriven	Undriven	Undriven	OFF
SD_VDD1_PWR_EN#	Primary	Undriven	Driven	Undriven	OFF
SD_3P3_RCOMP	Primary	Undriven	Undriven	Undriven	OFF
SD_1P8_RCOMP	Primary	Undriven	Undriven	Undriven	OFF
Note: *Reset reference for primary well pins is RSMRST#					

40.6 Functional Description

The SDXC controller handles SD Protocol at transmission, packing data, adding cyclic redundancy check (CRC), start/end bit, and checking for transaction format correctness. The SDXC main use case is to connect to an external detachable storage device. It supports SDXC card specification version 3.01. Both 1.8V and 3.3V signaling is supported. Additional information can be obtained from the SDXC 3.01 specification.

The following chart maps the working modes of SDXC.

Table 40-1. SD Working Modes

SDXC Mode	Data Rate	Clock Frequency	Maximum Data Throughput	Actual Throughput
Default Speed/SDR12	Single	0 – 25 MHz	12.5 MB/s	12 MB/s
High Speed/SDR25	Single	0 – 50 MHz	25 MB/s	24 MB/s
SDR50	Single	0 – 100 MHz	50 MB/s	48 MB/s
DDR50	Dual	0 – 50 MHz	50 MB/s	48 MB/s
SDR104	Single	0 – 208 MHz	104 MB/s	96 MB/s

40.7 Virtual GPIO (vGPIO) Used as Interrupt

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.

