Intel® Core™ Ultra 200H and 200U Series Processors

Datasheet, Volume 1 of 2

Rev. 002

May 2025

Doc. No.: 842704, Rev.: 002

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

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

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

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

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

Altering clock frequency, voltage, or memory interface speeds may void any product warranties and reduce stability, security, performance, and life of the processor and other components. Intel has not validated processor running memory above Plan-Of-Record (POR) speed. DRAM/DIMM devices should support desired speed, check with DRAM/DIMM vendors for details. System manufacturers are responsible for all validation and assume the risk of any stability, security, performance, or other functional issues resulting from such alterations

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

Copyright © 2025, Intel Corporation. All rights reserved.

Contents

Revision History....16

1.0 Introduction....17

1.1 Processor Volatility Statement....18
1.2 Package Support....18
1.3 Supported Technologies....19
1.3.1 API Support (Windows*)....22
1.3.2 Firmware Resiliency....22
1.4 Power Management Support....22
1.4.1 Processor Core Power Management....22
1.4.2 System Power Management....22
1.4.3 Memory Controller Power Management....22
1.4.4 Processor Graphics Power Management....23
1.5 Thermal Management Support....23
1.6 Ballout Information....24
1.7 Processor Testability....24
1.8 Operating Systems Support....24
1.9 Terminology and Special Marks....24
1.10 Flexible High Speed I/O....27
1.10.1 Processor-U....27
1.10.2 Processor-H....28
1.11 Related Documents....29

2.0 Processor and Device IDs....30

2.1 CPUID....30
2.2 PCI Configuration Header....30
2.3 Device IDs....31
2.4 Revision IDs....34

3.0 Package Mechanical Specifications....35

3.1 Package Mechanical Attributes....35
3.2 Package Loading and Tile Pressure Specifications....35
3.2.1 Static Compressive Load Specification....35
3.2.2 Maximum Pressure Specifications....36
3.3 Package Storage Specifications....37

4.0 Memory Mapping....38

4.1 Functional Description....38
4.1.1 PCI Devices and Functions....38
4.1.2 Fixed I/O Address Ranges....38
4.1.3 Variable I/O Decode Ranges....41
4.2 Memory Map....42
4.2.1 Boot Block Update Scheme....45

5.0 Security Technologies....47

5.1 Intel® Converged Boot Guard and Intel® TXT....47
5.2 Crypto Acceleration Instructions....48
5.2.1 Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)....48
5.2.2 Perform Carry-Less Multiplication Quad Word Instruction (PCLMULQDQ)....48

5.2.3 Intel® Secure Hash Algorithm Extensions (Intel® SHA Extensions).....49
5.2.4 New Cryptographic Acceleration Instructions.....49
5.3 Intel® Secure Key.....49
5.4 Execute Disable Bit 50
5.5 Intel® Supervisor Mode Execution Prevention (Intel® SMEP).....50
5.6 Intel® Supervisor Mode Access Prevention (Intel® SMAP).....50
5.7 User Mode Instruction Prevention (UMIP) 50
5.8 Read Processor ID (RDPID) 51
5.9 Intel® Total Memory Encryption - Multi-Key.....51
5.10 Control-flow Enforcement Technology (Intel® CET).....51
5.10.1 Shadow Stack.....52
5.10.2 Indirect Branch Tracking 52
5.11 Intel® System Resources Defense and Intel® System Security Report.....52
5.12 BIOS Guard.....53
5.13 Intel® Platform Trust Technology.....53
5.14 Linear Address Space Separation (LASS).....53
5.15 Security Firmware Engines.....53
5.15.1 Intel® Converged Security and Management Engine (Intel® CSME).....53
5.15.2 Intel® Silicon Security Engine.....53
5.15.3 Intel® Graphics System Controller (Intel® GSC).....54

6.0 Intel® Virtualization Technology (Intel® VT).....55
6.1 Intel® Virtualization Technology (Intel® VT) for Intel® 64 and Intel® Architecture (Intel® VT-x) 55
6.2 Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d) 57
6.3 Intel® APIC Virtualization Technology (Intel® APICv).....59

7.0 Platform Environmental Control Interface (PECI).....61
7.1 PECI Bus Architecture.....61

8.0 Intel® GMM and Neural Network Accelerator (Intel® GNA 3.0) 64

9.0 Intel® Image Processing Unit (Intel® IPU6).....66
9.1 Platform Imaging Infrastructure.....66
9.2 Intel® Image Processing Unit (Intel® IPU6).....67
9.3 Camera/MIPI.....67
9.3.1 Camera Pipe Support.....67
9.3.2 MIPI* CSI-2 Camera Interconnect.....67

10.0 Intel® Neural Processing Unit (Intel® NPU).....70
10.1 Functional Description.....70
10.1.1 Processor Subsystem.....71
10.1.2 NCE Subsystem.....72

11.0 Audio Voice and Speech.....74
11.1 Intel® High Definition Audio (Intel® HD Audio) Controller Capabilities.....75
11.2 Audio DSP Capabilities.....75
11.3 Intel® High Definition Audio Interface Capabilities.....76
11.4 Direct Attached Digital Microphone (PDM) Interface.....76
11.5 USB Audio Offload Support.....77
11.6 I^2S / PCM Interface.....77
11.7 Intel® Display Audio Interface.....77
11.8 MIPI® SoundWire* Interface.....78

11.9 Signal Description..... 78
11.10 Integrated Pull-Ups and Pull-Downs..... 81
11.11 I/O Signal Planes and States.....81

12.0 Power Management..... 83

12.1 System Power States, Advanced Configuration and Power Interface (ACPI) 83
12.2 Legacy Power Management Support.....86
12.2.1 ALT Access Mode.....86
12.2.2 Legacy Power Management Theory of Operation.....87
12.3 Functional Description.....87
12.3.1 Features..... 87
12.3.2 Power Saving Features.....88
12.3.3 SMI#/SCI Generation..... 89
12.3.4 Sleep States.....92
12.3.5 Event Input Signals and Their Usage.....94
12.3.6 System Power Supplies, Planes, and Signals..... 98
12.3.7 Reset Behavior.....101
12.4 Processor IA Core Power Management.....104
12.4.1 OS/HW Controlled P-states..... 104
12.4.2 Low Power Idle States.....104
12.4.3 Requesting the Low-Power Idle States.....105
12.4.4 Processor IA Core C-State Rules..... 105
12.4.5 Package C-States..... 106
12.4.6 Package C-States and Display Resolutions.....109
12.5 Processor Graphics Power Management 109
12.5.1 Memory Power Savings Technologies.....109
12.5.2 Display Power Savings Technologies.....110
12.5.3 Processor Graphics Core Power Savings Technologies..... 111
12.6 TCSS Power State.....112
12.7 Power and Performance Technologies..... 112
12.7.1 Intel® Thread Director 112
12.7.2 Intel® Smart Cache Technology.....113
12.7.3 P-core, E-core, and LP E-core Level 0, Level 1 and Level 2 Caches..... 113
12.7.4 Ring Interconnect.....116
12.7.5 Intel® Hybrid Technology.....116
12.7.6 Intel® Hyper-Threading Technology.....117
12.7.7 Intel® Turbo Boost Technology 2.0.....117
12.7.8 Intel® Turbo Boost Max Technology 3.0..... 118
12.7.9 Intel® Adaptive Boost Technology..... 118
12.7.10 Intel System Agent Enhanced SpeedStep ® Technology 119
12.7.11 Enhanced Intel SpeedStep® Technology.....119
12.7.12 Intel® Speed Shift Technology 120
12.7.13 Intel® Advanced Vector Extensions 2 (Intel® AVX2) 120
12.7.14 Intel® 64 Architecture x2APIC..... 120
12.7.15 Intel® Dynamic Tuning Technology (Intel® DTT) 122
12.7.16 Cache Line Write Back (CLWB).....122
12.7.17 User Mode Wait Instructions 122
12.8 Deprecated Technology..... 122
12.9 Power and Internal Signals..... 123
12.9.1 Signal Description.....123
12.9.2 Power Sequencing Signals..... 124

12.9.3 Integrated Pull-Ups and Pull-Downs..... 125
12.9.4 I/O Signal Planes and States..... 125

13.0 Power Delivery..... 127
13.1 Power and Ground Signals..... 127
13.2 Digital Linear Voltage Regulator (DLVR)..... 128
13.3 Fast V-Mode (FVM)..... 128
13.4 Current Excursion Protection (CEP)..... 128
13.5 Reactive PL4 with PL4 Boost..... 128

14.0 Electrical Specifications..... 130

15.0 Thermal Management..... 131
15.1 Processor Thermal Management..... 131
15.1.1 Thermal Considerations..... 131
15.1.2 Thermal Management Features..... 134
15.1.3 Assured Power (cTDP)..... 141
15.1.4 Intel® Memory Thermal Management 142
15.2 Processor Base Power Thermal and Power Specifications 143
15.3 Processor Thermal and Power Specifications..... 145
15.4 Error and Thermal Protection Signals..... 147
15.5 Thermal Sensor..... 147
15.5.1 Modes of Operation..... 147
15.5.2 Temperature Trip Point..... 148
15.5.3 Thermal Sensor Accuracy (Taccuracy)..... 148
15.5.4 Thermal Reporting to EC..... 148
15.5.5 Thermal Trip Signal (SOCHOT#)..... 148

16.0 System Clocks..... 149
16.1 Integrated Clock Controller (ICC)..... 149
16.1.1 Signal Description..... 149
16.2 I/O Signal Pin States..... 150
16.3 Clock Topology..... 151
16.3.1 Integrated Reference Clock PLL..... 151

17.0 Real Time Clock (RTC)..... 152
17.1 Signal Description..... 152
17.2 I/O Signal Planes and States..... 153

18.0 Memory..... 154
18.1 System Memory Interface..... 154
18.1.1 Processor SKU Support Matrix..... 154
18.1.2 Supported Memory Modules and Devices..... 155
18.1.3 System Memory Timing Support..... 156
18.1.4 Memory Controller (MC)..... 158
18.1.5 System Memory Controller Organization Mode..... 158
18.1.6 System Memory Frequency..... 160
18.1.7 Technology Enhancements of Intel® Fast Memory Access (Intel® FMA)..... 160
18.1.8 Data Scrambling..... 160
18.1.9 Data Swapping 161
18.1.10 LPDDR5/x CMD/ADD Ascending and Descending 161
18.1.11 DDR I/O Interleaving..... 161
18.1.12 DRAM Clock Generation 161

18.1.13 DRAM Reference Voltage Generation 162
18.1.14 Data Swizzling.....162
18.1.15 Error Correction With Standard RAM.....162
18.1.16 Post Package Repair (PPR)..... 162
18.2 Integrated Memory Controller (IMC) Power Management..... 162
18.2.1 Disabling Unused System Memory Outputs..... 162
18.2.2 DRAM Power Management and Initialization..... 163
18.2.3 DDR Electrical Power Gating.....164
18.2.4 Power Training..... 164
18.3 Signal Description..... 165

19.0 USB Type-C* Sub System..... 168
19.1 General Capabilities.....168
19.2 USB4* Router.....170
19.2.1 USB4 Host Router Implementation Capabilities.....170
19.3 xHCI/xDCI Controllers 171
19.3.1 USB 3 Controllers.....171
19.3.2 PCIe Interface.....172
19.4 Display Interface.....172
19.5 USB Type-C Signals..... 172
19.6 AUX BIAS Control.....172

20.0 Universal Serial Bus (USB).....175
20.1 Functional Description.....175
20.1.1 eXtensible Host Controller Interface (xHCI) Controller..... 175
20.1.2 USB Dual Role Support - eXtensible Device Controller Interface (xDCI) Controller.....175
20.2 Signal Description..... 176
20.3 Integrated Pull-Ups and Pull-Downs..... 178
20.4 I/O Signal Planes and States.....178
20.5 Supported USB 2.0 Ports.....179

21.0 Intel® Volume Management Device (Intel® VMD) Technology 180

22.0 PCI Express* (PCIe*)..... 182
22.1 Functional Description.....182
22.1.1 PCI Express* Power Management..... 184
22.1.2 Port 80h Decode.....184
22.1.3 Separate Reference Clock with Independent SSC (SRIS).....185
22.1.4 Advanced Error Reporting.....185
22.1.5 Single - Root I/O Virtualization (SR - IOV)..... 185
22.1.6 PCI Express* Receiver Lane Polarity Inversion..... 185
22.1.7 Precision Time Measurement (PTM) 186
22.2 Signal Description..... 186
22.3 I/O Signal Planes and States.....187
22.4 PCI Express* Root Port Support Feature Details.....187

23.0 Serial ATA (SATA)..... 191
23.1 Functional Description.....191
23.1.1 Features Supported..... 191
23.1.2 SATA 6 Gb/s Support..... 192
23.1.3 Hot Plug Operation..... 192

23.1.4 Intel® Rapid Storage Technology (Intel® RST)....192
23.1.5 Power Management Operation....194
23.1.6 SATA Device Presence....196
23.1.7 SATA LED....196
23.1.8 Advanced Host Controller Interface (AHCI) Operation....196
23.2 Signal Description....197
23.3 Integrated Pull-Ups and Pull-Downs....198
23.4 I/O Signal Planes and States....198

24.0 Graphics....200
24.1 Processor Graphics....200
24.1.1 Media Support (Intel® QuickSync and Clear Video Technology HD)....200
24.1.2 Graphics Core Cache....203
24.2 Platform Graphics Hardware Feature....203
24.2.1 Hybrid Graphics....203

25.0 Display....204
25.1 Display Technologies Support....204
25.2 Display Interfaces....204
25.2.1 Digital Display Interface DDI Signals....205
25.2.2 Digital Display Interface TCP Signals....206
25.3 Display Features....207
25.3.1 General Capabilities....208
25.3.2 Multiple Display Configurations....209
25.3.3 High-bandwidth Digital Content Protection (HDCP)....209
25.3.4 DisplayPort*....209
25.3.5 High-Definition Multimedia Interface (HDMI*)....211
25.3.6 embedded DisplayPort* (eDP*)....213
25.3.7 Integrated Audio....213

26.0 Processor Sideband Signals....215
26.1 Signal Description....215
26.2 Integrated Pull-Ups and Pull-Downs....215
26.3 I/O Signal Planes and States....215

27.0 General Purpose Input and Output....216
27.1 Functional Description....216
27.1.1 Interrupt / IRQ via GPIO Requirement....216
27.1.2 Integrated Pull-ups and Pull-downs....216
27.1.3 SCI / SMI# and NMI....216
27.1.4 Timed GPIO....217
27.1.5 GPIO Blink (BK) and Serial Blink (SBK)....217
27.1.6 GPIO Ownership....218
27.1.7 Native Function and TERM Bit Setting....218
27.2 Signal Description....218

28.0 Interrupt Timer Subsystem (ITSS)....219
28.1 Feature Overview....219
28.2 Functional Description....219
28.2.1 8254 Timers....220
28.2.2 APIC Advanced Programmable Interrupt Controller....222
28.2.3 High Precision Event Timer (HPET)....222

29.0 GPIO Serial Expander....227
29.1 Functional Description....227
29.2 Signal Description....228
29.3 Integrated Pull-ups and Pull-downs....228
30.0 Intel® Serial I/O Inter-Integrated Circuit (I^2C) Controllers....229
30.1 Functional Description....230
30.1.1 Protocols Overview....230
30.1.2 DMA Controller....231
30.1.3 Reset....232
30.1.4 Power Management....232
30.1.5 Interrupts....232
30.1.6 Error Handling....233
30.1.7 Programmable SDA Hold Time....233
30.2 Signal Description....233
30.3 Integrated Pull-Ups and Pull-Downs....234
30.4 I/O Signal Planes and States....234
31.0 Intel® Serial I/O Improved Inter-Integrated Circuit (I^3C) Controllers....235
31.1 Functional Description....236
31.1.1 Reset....236
31.1.2 Power Management....236
31.1.3 Interrupts....237
31.2 Signal Description....237
31.3 Integrated Pull-Ups and Pull-Downs....237
31.4 I/O Signal Planes and States....237
32.0 Gigabit Ethernet Controller....238
32.1 Functional Description....238
32.1.1 GbE PCI Bus Interface....240
32.1.2 Error Events and Error Reporting....241
32.1.3 Ethernet Interface....241
32.1.4 PCI Power Management....241
32.2 Signal Description....242
32.3 Integrated Pull-Ups and Pull-Downs....243
32.4 I/O Signal Planes and States....243
33.0 Connectivity Integrated (CNVi)....244
33.1 Functional Description....244
33.2 Signal Description....245
33.3 Integrated Pull-ups and Pull-downs....247
33.4 I/O Signal Planes and States....248
34.0 Controller Link....249
34.1 Signal Description....249
34.2 Integrated Pull-Ups and Pull-Downs....249
34.3 I/O Signal Planes and States....250
34.4 External CL_RST# Pin Driven/Open-drained Mode Support....250
35.0 Integrated Sensor Hub (ISH)....251
35.1 Features....252
35.1.1 ISH I^2C Controllers....252

35.1.2 ISH I3C Controllers....252
35.1.3 ISH UART Controller....253
35.1.4 ISH GSPI Controller....253
35.1.5 ISH GPIOs....253
35.2 Functional Description....253
35.2.1 ISH Micro-Controller....253
35.2.2 SRAM....253
35.2.3 PCI Host Interface....254
35.2.4 ISH IPC....254
35.2.5 ISH Interrupt Handling via IOAPIC (Interrupt Controller)....255
35.3 Signal Description....255
35.4 Integrated Pull-Ups and Pull-Down....256
35.5 I/O Signal Planes and States....256

36.0 System Management Interface and SMLink....258
36.1 Functional Description....258
36.1.1 Integrated USB-C* Usage....258
36.2 Signal Description....259
36.3 Integrated Pull-Ups and Pull-Downs....260
36.4 I/O Signal Planes and States....260
36.5 System Management....260
36.5.1 Theory of Operation....261

37.0 Host System Management Bus (SMBus) Controller....264
37.1 Functional Description....264
37.1.1 Host Controller....264
37.1.2 SMBus Target Interface....271
37.2 SMBus Power Gating....278
37.3 Signal Description....278
37.4 Integrated Pull-Ups and Pull-Downs....278
37.5 I/O Signal Planes and States....279

38.0 Serial Peripheral Interface (SPI)....280
38.1 Functional Description....280
38.1.1 SPI0 for Flash....280
38.1.2 SPI0 Support for TPM....288
38.2 Signal Description....288
38.3 Integrated Pull-Ups and Pull-Downs....288
38.4 I/O Signal Planes and States....289

39.0 Enhanced Serial Peripheral Interface (eSPI)....290
39.1 Functional Description....290
39.1.1 Operating Frequency....290
39.1.2 WAIT States from eSPI device....291
39.1.3 In-Band Link Reset....291
39.1.4 Flash Sharing Mode....291
39.1.5 PECI Over eSPI....291
39.1.6 Multiple OOB processes....291
39.1.7 Channels and Supported Transactions....291
39.2 Signal Description....297
39.3 Integrated Pull-Ups and Pull-Downs....297
39.4 I/O Signal Planes and States....297

40.0 Intel® Serial IO Generic SPI (GSPI) Controllers..... 299
40.1 Functional Description.....299
40.1.1 Controller Overview..... 299
40.1.2 DMA Controller.....300
40.1.3 Reset.....301
40.1.4 Power Management..... 301
40.1.5 Interrupts.....302
40.1.6 Error Handling..... 302
40.2 Signal Description..... 302
40.3 Integrated Pull-Ups and Pull-Downs..... 303
40.4 I/O Signal Planes and States.....303

41.0 Touch Host Controller (THC)..... 304
41.1 Functional Description.....304
41.2 Signal Description..... 305
41.3 Integrated Pull-Ups and Pull-Downs..... 306
41.4 I/O Signal Planes and States.....306

42.0 Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers.....307
42.1 Functional Description.....308
42.1.1 UART Serial (RS-232) Protocols Overview..... 308
42.1.2 16550 8-bit Addressing - Debug Driver Compatibility..... 309
42.1.3 DMA Controller.....309
42.1.4 Reset.....310
42.1.5 Power Management 310
42.1.6 Interrupts.....311
42.1.7 Error Handling..... 311
42.2 Signal Description..... 311
42.3 Integrated Pull-Ups and Pull-Downs..... 312
42.4 I/O Signal Planes and States.....312
42.5 LSx..... 312
42.5.1 LSx Signal Description.....313
42.5.2 Integrated Pull-Ups and Pull-Downs..... 313
42.5.3 I/O Signal Planes and States.....313

43.0 Private Configuration Space Port ID.....314

44.0 Testability and Monitoring..... 316
44.1 Signal Description..... 316
44.2 Intel® Processor Trace..... 317
44.3 I/O Signal Planes and States.....318

45.0 Miscellaneous Signals..... 319
45.1 Signal Description..... 319
45.2 Integrated Pull-Ups and Pull-Downs..... 320
45.3 Ground and Reserved Signals.....320

Figures

1 H/U Processor Line Platform Diagram.... 18
2 Processor-U Flexible HSIO Lane Details 27
3 Flexible HSIO Lane Details 28
4 Device to Domain Mapping Structures 58
5 PECI Host-Clients Connection Example.... 62
6 PECI EC Connection Example.... 63
7 Processor Camera System.... 66
8 NPU IP Block Diagram 71
9 Power State Block Diagram....85
10 Power Management Substates....89
11 Idle Power Management Breakdown of the Processor IA Cores.... 105
12 U-Processor P-core, E-core, and LP E-core Cache Hierarchy.... 114
13 H-Processor P-core, E-core, and LP E-core Cache Hierarchy.... 116
14 Package Power Control.... 133
15 PROCHOT Demotion Description 138
16 ICC Diagram.... 149
17 Intel® DDR5 Flex Memory Technology Operations.... 159
18 GPIO - Virtual Wire Index Bit Mapping 173
19 Supported USB 2.0 Ports on H-Processor.... 179
20 Technology Description....180
21 Processor-U Supported PCI Express* Link Configurations 188
22 Processor-H Supported PCI Express* Link Configurations 189
23 Port Enable/Device Present Bits Flow....196
24 Processor Display Architecture....208
25 DisplayPort* Overview....210
26 HDMI* Overview 212
27 GSX Topology - Example....227
28 Data Transfer on I^2C Bus.... 230
29 TCO Compatible Mode SMBus Configuration....261
30 Advanced TCO Mode....263
31 Flash Descriptor Regions....283
32 Flash Descriptor Redundancy....286
33 eSPI Device Request to Processor for Processor Temperature....294
34 Processor Response to eSPI device with Processor Temperature 294
35 eSPI Device Request to Processor for Processor RTC Time.... 295
36 Processor Response to eSPI device with RTC Time 296
37 THC Block Diagram.... 305
38 UART Serial Protocol 308
39 UART Receiver Serial Data Sample Points....309

Tables

1 Processor Series 17
2 Terminology 24
3 Special Marks 27
4 Acronyms 27
5 CPUID Format 30
6 PCI Configuration Header 31
7 Host Device ID (DID0) and Processor Graphics Device ID (DID2) 31
8 Other Device ID 31
9 ACPI Device ID for GPIO Controller 34
10 H/U Package Mechanical Attributes 35
11 Package Loading Specifications 36
12 Fixed I/O Ranges Decoded by Processor 38
13 Variable I/O Decode Ranges 41
14 Processor Memory Decode Ranges (Processor Perspective) 42
15 Boot Block Update Scheme 45
16 Acronyms 74
17 References 75
18 Integrated Pull-Ups and Pull-Downs 81
19 I/O Signal Planes and States 81
20 Acronyms 83
21 References 83
22 General System Power States 83
23 State Transition Rules for the Processor 84
24 System Power Plane 84
25 Write Only Registers with Read Paths in ALT Access Mode 86
26 PIC Reserved Bits Return Values 87
27 Causes of SMI and SCI 90
28 Sleep Types 92
29 Causes of Wake Events 93
30 Transitions Due to Power Failure 94
31 Transitions Due to Power Button 95
32 PRIMPWRDNACK/GPP_A02 Pin Behavior 101
33 PRIMPWRDNACK During Reset 101
34 Causes of Host and Global Resets 102
35 Core C-states 106
36 Package C-States 107
37 Deepest Package C-State Available 109
38 TCSS Power State 112
39 Power Sequencing Signals 124
40 H and U - Processor Series Power Rail Description 127
41 H and U - Processor Series Power Rail Sense Signals 127
42 Definitions/Acronyms 131
43 Assured Power 141
44 General Notes 143
45 Processor Base Power (TDP) Specifications (H-Processor) 144
46 Processor Base Power Specifications (U-Processor) 145
47 Package Turbo Specifications (H/U -Processors) 145
48 Processor Temperature Specifications (H/U- Processors) 146
49 Error and Thermal Protection Signals 147
50 Signal Description 149
51 I/O Signal Pin States 150
52 Acronyms 152
53 DDR Support Matrix Table 154

54 Supported DDR5 Non-ECC SoDIMM/CSoDIMM Module Configurations (H/U-Processor Series)..... 155
55 Supported DDR5 Memory Down Device Configurations (H/U - Processor Series) 155
56 Supported LPDDR5/x x32 DRAMs Configurations (H/U - Processor Series) 155
57 Supported LPDDR5/x x64 DRAMs Configurations (H/U - Processor Series)..... 156
58 DDR System Memory Timing Support..... 156
59 LPDDR System Memory Timing Support 157
60 SA Speed Enhanced Speed Steps (SA-GV) and Gear Mode Frequencies 157
61 LPDDR5/x CMD/ADD Ascending and Descending..... 161
62 DDR5 Memory Interface..... 165
63 LPDDR5/x Memory Interface..... 165
64 USB Type-C* Port Configuration..... 169
65 USB Type-C* Lanes Configuration..... 169
66 USB Type-C* Non-Supported Lane Configuration..... 170
67 PCIe via USB4 Configuration..... 172
68 Acronyms..... 175
69 Acronym..... 182
70 Reference Table..... 182
71 Features Supported..... 182
72 Power Plane and States for PCI Express* Signals 187
73 PCI Express* Root Port Feature Details 187
74 Acronyms..... 191
75 References..... 191
76 Hardware Accelerated Video Decoding 201
77 Hardware Accelerated Video Encode 201
78 Display Ports Availability and Link Rate..... 204
79 Digital Display Interface DDI Signals..... 205
80 Digital Display Interface TCP Signals..... 206
81 Display Resolutions and Link Bandwidth for Multi-Stream Transport Calculations..... 210
82 DisplayPort Maximum Resolution..... 211
83 HDMI Maximum Resolution..... 212
84 Embedded DisplayPort Maximum Resolution..... 213
85 Processor Supported Audio Formats over HDMI* and DisplayPort*..... 213
86 Acronyms..... 215
87 Acronyms..... 216
88 Native Function Signals Supporting Dynamic Termination Override..... 218
89 Acronyms..... 219
90 References..... 219
91 Counter Operating Modes..... 221
92 Acronyms..... 230
93 References..... 230
94 Acronyms..... 236
95 Acronyms..... 238
96 References..... 238
97 LAN Mode Support..... 241
98 GbE LAN Signals..... 242
99 Acronyms..... 244
100 References..... 244
101 Acronyms..... 249
102 Acronyms..... 251
103 References..... 252
104 IPC Initiator -> Target flows..... 254
105 Acronyms..... 258
106 Acronyms..... 261
107 Event Transitions that Cause Messages..... 262

108 Acronyms.... 264
109 References....264
110 I^2C* Block Read....268
111 Enable for SMBALERT# 270
112 Enables for SMBus Target Write and SMBus Host Events....270
113 Enables for the Host Notify Command....270
114 Target Write Registers.... 272
115 Command Types....272
116 Target Read Cycle Format.... 273
117 Data Values for Target Read Registers....273
118 Host Notify Format....275
119 Target Read Cycle Format 276
120 Data Values for Target Read Registers....276
121 Enables for SMBus Target Write and SMBus Host Events....278
122 Acronyms.... 280
123 SPI0 Flash Regions....281
124 Region Size Versus Erase Granularity of Flash Components 281
125 Region Access Control Table.... 284
126 Flash Descriptor Processor Complex Soft Strap.... 285
127 Acronyms.... 290
128 References....290
129 eSPI Channels and Supported Transactions.... 292
130 eSPI Virtual Wires (VW)....292
131 Acronyms.... 299
132 Acronyms.... 304
133 Acronyms.... 308
134 Private Configuration Space Register Target Port IDs 314
135 Acronyms.... 316
136 References....316
137 Testability Signals.... 316
138 Power Planes and States for Testability Signals.... 318
139 Signal Descriptions.... 319
140 Integrated Pull-Ups and Pull-Downs....320
141 GND, RSVD, and NCTF Signals....321

Revision History

Document Number	Revision Number	Description	Revision Date
842704	001	Initial Release	January 2025
842704	002	Power Management on page 83 • **Added** note in Intel® Turbo Boost Technology 2.0 on page 117, Intel® Hyper-Threading Technology on page 117 • **Removed** AVX-512 and FP16 support from Intel® Hybrid Technology on page 116 and Intel® Advanced Vector Extensions 2 (Intel® AVX2) on page 120	May 2025

1.0 Introduction

This document is intended for Original Equipment Manufacturers (OEMs), Original Design Manufacturers (ODM) and BIOS vendors creating products based on the Intel® Core™ Ultra 200H and 200U Series Processors.

This document 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 document abbreviates buses as Bn, devices as Dn and functions as Fn. 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.

The Intel® Core™ Ultra Processors are 64-bit, multi-core processors built on N3B process technology.

- The H-Processor Line offered in a 1-Chip Platform that includes the Compute, SOC, GT, and IOE tiles on the same package.
- The U-Processor Line offered in a 1-Chip Platform that includes the Compute, SOC, GT, and IOE tiles on the same package.

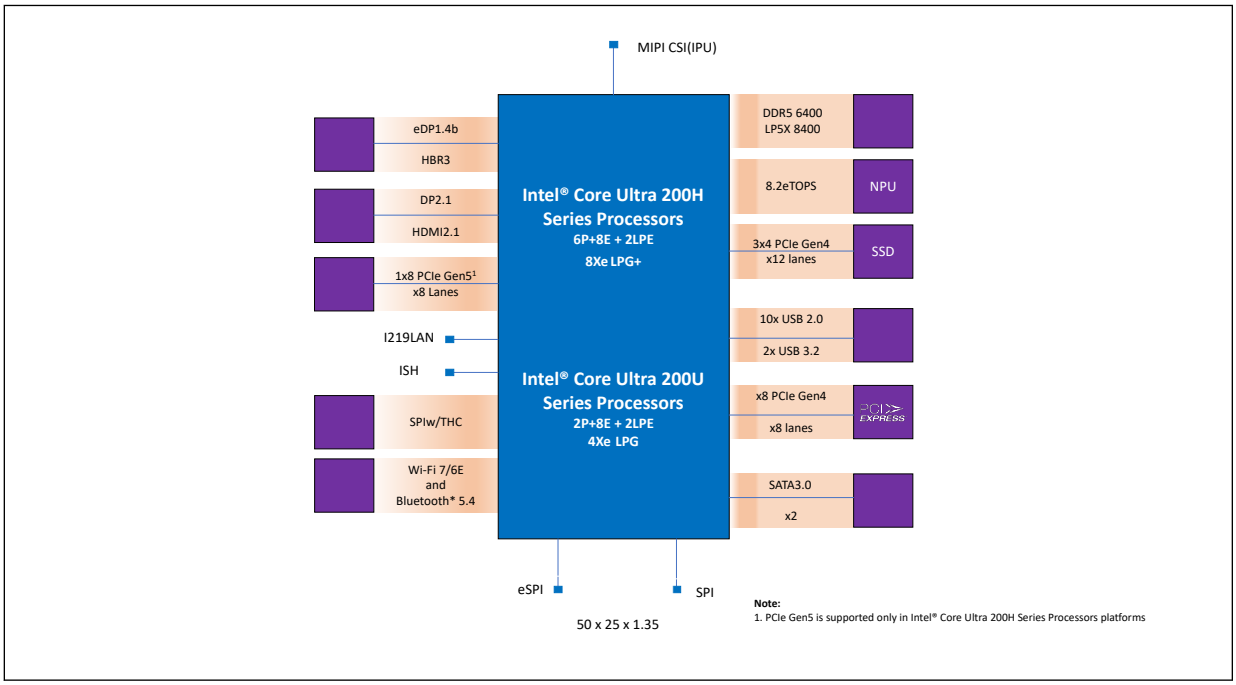
The following table describes the different H/U processor lines:

Table 1. Processor Series

Processor Line[1]	Package	Processor Base Power (a.k.a. TDP)[2,3]	Compute Tile P-cores	Compute Tile E-cores	SOC Tile E-cores	Graphics Configuration Xe-cores	Platform Type
H BGA	BGA2049	45 W	up to 6	8	2	up to 8	1-Chip
H BGA	BGA2049	28 W	up to 6	8	2	up to 8	1-Chip
U BGA	BGA2049	15 W	2	8	2	4	1-Chip

Notes: 1. Processor series offering may change.
2. For adjustment to the Processor Base Power (TDP) it is required to preserve base frequency associated with the sustained long-term thermal capability.
3. Processor Base Power (a.k.a. TDP) workload does not reflect I/O connectivity cases such as Thunderbolt.

Figure 1. H/U Processor Line Platform Diagram



NOTE

- Not all processor interfaces and features are presented in all Processor Series. The presence of various interfaces and features will be indicated within the relevant sections and tables.

  H processor does not support the Intel® Hyper-Threading Technology (Intel® HT Technology).

1.1 Processor Volatility Statement

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

NOTE

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

1.2 Package Support

The Intel® Core™ Ultra H and U Processors are available in the following package:

BGA2049

- A 25 x 50 mm
- Substrate Z = 0.644 ± 0.095 mm
- 1.245 ± 0.109 mm (BOTTOM OF BGA TO TOP OF TILE)

1.3 Supported Technologies

- PCI Express* (PCIe*)
- Flexible High Speed I/O
- Xe Graphics Core Based Processor Graphics
- Display
  - DisplayPort* 2.1 (DP* 2.1)
  - Embedded DisplayPort* 1.4 (eDP* 1.4)
  - High-Definition Multimedia Interface* 2.1 (HDMI* 2.1)
- Intel® Gaussian & Neural Accelerator 3.5 (Intel® GNA 3.5)
- Intel® Image Processing Unit (Intel® IPU 6)
- Intel® Neural Processing Unit (Intel® NPU)
- USB Type-C* Sub System
  - Intel® Thunderbolt™
  - USB 3.2 Gen 1x1 (5 Gb/s) Dual Role (eXtensible Device Controller - xDCI) Capability
  - USB 3.2 Gen 2x1 (10 Gb/s) eXtensible Host Controller (xHCI)
  - USB 3.2 Gen 2x2 (20 Gb/s) eXtensible Host Controller (xHCI)
  - DisplayPort* 2.1 (DP* 2.1)
- Memory
  - DDR5
  - LPDDR5x
  - In Band Error Code Correction (IBECC)
- Platform Environmental Control Interface (PECI)
- Universal Serial Bus (USB)
- Intel® Volume Management Device (Intel® VMD)
- Touch Host Controller (THC)
- Intel® Serial I/O I2C and I3C Host controllers
- Integrated Sensor Hub (ISH)
- Integrated Connectivity (CNVi)
- Intel® Integrated Serial ATA Host controller 3.2 (Intel® SATA 3.2)
- Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers
- Intel® Serial IO Generic SPI (GSPI) Controllers
- Serial Peripheral Interface (SPI)
- Enhanced Serial Peripheral Interface (eSPI)
- Intel® High Definition Audio (Intel® HD Audio)
- Intel® Smart Sound Technology (Intel® SST)
- System Management Bus (SMBus)
- Gigabit Ethernet (GbE) controller

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

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

NOTE

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

1.3.1 API Support (Windows*)

- Direct3D 12.2, Direct3D 12.1, Direct3D 12, , Direct3D 11.4, Direct3D 11.3, Direct3D 11.2, Direct3D 11.1, Direct3D 10.1, Direct3D 10, Direct3D 9.0L via DX9on12, Direct3D 9.0C via DX9on12, Direct2D
- OpenGL* 4.6
- Open CL* 3.0
- Vulkan 1.2

DirectX* extensions:

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

Gen 12.7 architecture delivers hardware acceleration of Direct X* 12.2 Render pipeline comprising of the following stages: Vertex Fetch, Vertex Shader, Hull Shader, Tessellation, Domain Shader, Geometry Shader, Rasterizer, Pixel Shader, Pixel Output, Raytracing, Mesh shading, Variable rate shading, Sampler feedback.

1.3.2 Firmware Resiliency

Intel's NVMe based recovery supports recovery of all firmware on Intel® Core™ Ultra Processor from NVMe storage boot partition in a secure manner.

Firmware Resiliency and Recovery in-field is critical to keep PCs up and running while preventing the requirement of additional space on SPI flash to keep a backup firmware. Therefore, it decreases the Platform BOM cost.

1.4 Power Management Support

1.4.1 Processor Core Power Management

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

- C0, C2, C3, C6, C8, and C10

Refer to Processor IA Core Power Management on page 104 for more information.

1.4.2 System Power Management

Supports the following power management system states:

- Modern Standby (S0ix)
- S4
- S5

Refer to Power Management on page 83 for more information.

1.4.3 Memory Controller Power Management

- Disabling Unused System Memory Outputs

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

Refer to Integrated Memory Controller (IMC) Power Management on page 162 for more information.

1.4.4 Processor Graphics Power Management

Memory Power Savings Technologies

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

Display Power Savings Technologies

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

Graphics Core Power Savings Technologies

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

1.5 Thermal Management Support

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

Refer to Thermal Management on page 131 for more information.

1.6 Ballout Information

For information on U/H processor ball information, download the pdf, click on the navigation pane and refer the spreadsheet, **842704-001_U_H_Ballout.xlsx**.

1.7 Processor Testability

A DCI on-board connector should be placed to enable Intel® Core™ Ultra 200H and 200U Series Processors full debug capabilities. For Intel® Core™ Ultra 200H and 200U Series Processor SKUs, a Direct Connect Interface Tool connector is highly recommended to enable lower C-state to debug.

1.8 Operating Systems Support

Processor Series	Windows* 10 OS	Windows* 11 OS	Chrome* OS	Linux* OS
H	Yes	Yes	No	Yes
U	Yes	Yes	No	Yes

NOTE

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

1.9 Terminology and Special Marks

Table 2. Terminology

Term	Description
4K	Ultra High Definition (UHD)
AES	Advanced Encryption Standard
AGC	Adaptive Gain Control
API	Application Programming Interface
AVC	Advanced Video Coding
BLT	Block Level Transfer
BPP	Bits per Pixel
CDR	Clock and Data Recovery
CTLE	Continuous Time Linear Equalizer
D0ix-states	USB controller power states ranging from D0i0 to D0i3, where D0i0 is fully powered on and D0i3 is primarily powered off. Controlled by SW.
DDC	Digital Display Channel
DDI	Digital Display Interface for DisplayPort or HDMI/DVI
DSI	Display Serial Interface
DDR5	Firth-Generation Double Data Rate SDRAM Memory Technology

continued...

Term	Description
DFE	Decision Feedback Equalizer
DMA	Direct Memory Access
DPPM	Dynamic Power Performance Management
DP*	DisplayPort*
DSC	Display Stream Compression
DSI	Display Serial Interface
DTS	Digital Thermal Sensor
ECC	Error Correction Code - used to fix DDR transactions errors
eDP*	Embedded DisplayPort*
EU	Execution Unit in the Graphics Processor
GSA	Graphics in System Agent
GNA	Gaussian & Neural-Network Accelerator
HDCP	High-Bandwidth Digital Content Protection
HDMI*	High Definition Multimedia Interface
IMC	Integrated Memory Controller
Intel® 64 Technology	64-bit memory extensions to the IA-32 architecture
Intel® DPST	Intel® Display Power Saving Technology
Intel® PTT	Intel® Platform Trust Technology
Intel® TXT	Intel® Trusted Execution Technology
Intel® VT	Intel® Virtualization Technology. Processor Virtualization, when used in conjunction with Virtual Machine Monitor software, enables multiple, robust independent software environments inside a single platform.
Intel® VT-d	Intel® Virtualization Technology (Intel® VT) for Directed I/O. Intel® VT-d is a hardware assist, under system software (Virtual Machine Manager or OS) control, for enabling I/O device Virtualization. Intel® VT-d also brings robust security by providing protection from errant DMAs by using DMA remapping, a key feature of Intel® VT-d.
Intel® TH	Intel® Trace Hub
IOV	I/O Virtualization
IPU	Image Processing Unit
LFM	Low Frequency Mode. corresponding to the Enhanced Intel SpeedStep® Technology's lowest voltage/frequency pair. It can be read at MSR CEh [47:40].
LLC	Last Level Cache
LPDDR5/x	Low Power Double Data Rate SDRAM memory technology /x.
LPSP	Low-Power Single Pipe
LSF	Lowest Supported Frequency.This frequency is the lowest frequency where manufacturing confirms logical functionality under the set of operating conditions.
LTR	The Latency Tolerance Reporting (LTR) mechanism enables Endpoints to report their service latency requirements for Memory Reads and Writes to the Root Complex, so that power management policies for central platform resources (such as main memory, RC internal interconnects, and snoop resources) can be implemented to consider Endpoint service requirements.

continued...

Term	Description
MFM	Minimum Frequency Mode. MFM is the minimum ratio supported by the processor and can be read from MSR CEh [55:48].
MLC	Mid-Level Cache
MPEG	Motion Picture Expert Group, international standard body JTC1/SC29/WG11 under ISO/IEC that has defined audio and video compression standards such as MPEG-1, MPEG-2, and MPEG-4, etc.
NCTF	Non-Critical to Function. NCTF locations are typically redundant ground or non-critical reserved balls/lands, so the loss of the solder joint continuity at end of life conditions will not affect the overall product functionality.
PECI	Platform Environment Control Interface
PEG	PCI Express* Graphics
PL1, PL2, PL3	Power Limit 1, Power Limit 2, Power Limit 3
PMIC	Power Management Integrated Circuit
Processor	The 64-bit multi-core component (package)
Processor Core	The term "processor core" refers to the Si tile itself, which can contain multiple execution cores. Each execution core has an instruction cache, data cache, and 256-KB L2 cache. All execution cores share the LLC.
Processor Graphics	Intel® Processor Graphics
PSR	Panel Self-Refresh
PSx	Power Save States (PS0, PS1, PS2, PS3, PS4)
Rank	A unit of DRAM corresponding to four to eight devices in parallel, ignoring ECC. These devices are usually, but not always, mounted on a single side of a SoDIMM.
S0ix-states	Processor residency idle standby power states.
SCI	System Control Interrupt. SCI is used in the ACPI protocol.
SDP	Scenario Design Power
SHA	Secure Hash Algorithm
SSC	Spread Spectrum Clock
STR	Suspend to RAM
TAC	Thermal Averaging Constant
TBT	Thunderbolt™ Interface
TCC	Thermal Control Circuit
TDP	Processor Base Power (a.k.a. Thermal Design Power)
TTV Processor Base Power (a.k.a. TDP)	Thermal Test Vehicle Processor Base Power (a.k.a. Thermal Design Power)
V_{CC}	Processor Core Power Supply
V_{CCGT}	Processor Graphics Power Supply
V_{CCSA}	System Agent Power Supply
VLD	Variable Length Decoding
VPID	Virtual Processor ID
V_{SS}	Processor Ground

Table 3. Special Marks

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

1.10 Flexible High Speed I/O

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

NOTE

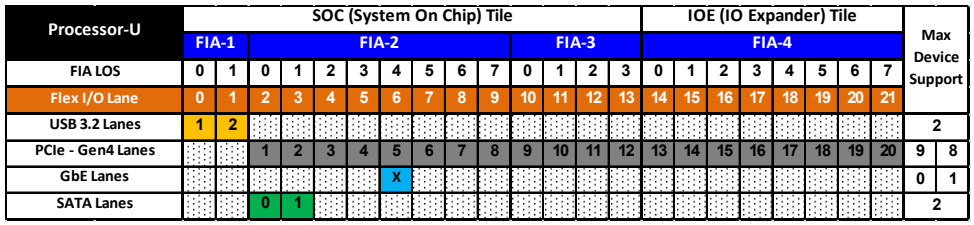
Some Flexible I/O multiplexing capabilities are not available on all SKUs. Refer to Introduction on page 17 for specific SKU implementation details.

Table 4. Acronyms

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

1.10.1 Processor-U

Figure 2. Processor-U Flexible HSIO Lane Details



FIA = Flex-IO Adapter

FIA LOS = Flex-IO Adapter Lane Ownership Number

The 22 Flexible HSIO Lanes [21:0] support the following:

1. Up to twenty PCIe* Lanes
   - A maximum of nine PCIe* Root Ports (or devices) can be enabled when GbE Port is disabled
   - A maximum of eight PCIe* Root Ports (or devices) can be enabled when GbE Port is enabled
   - PCIe* Lanes 1-4 (PCIe* Controller #1 Gen4), 5-8 (PCIe* Controller #2 Gen4), 9-12 (PCIe* Controller #3 Gen4), 13-16 (PCIe* Controller #4 Gen4), and 17-20 (PCIe* Controller #5 Gen4) must be individually configured
2. Up to two USB 3.2 Gen 1x1/2x1 Lanes
   - A maximum of two USB 3.2 Gen 1x1/2x1 Ports (or devices) can be enabled
   - USB 3.2 Gen 1x1 = First Generation with One 5 GT/s Data Lane
   - USB 3.2 Gen 2x1 = Second Generation with One 10 GT/s Data Lane
3. Up to two SATA Lanes
   - A maximum of two SATA Ports (or devices) can be enabled
4. Up to one GbE Lane
   - A maximum of one GbE Port can be enabled

1.10.2 Processor-H

Figure 3. Flexible HSIO Lane Details

Processor-H	SOC (System On Chip) Tile														IOE (IO Expander) Tile																Max Device Support	
	FIA-1		FIA-2								FIA-3				FIA-4								FIA-5									
FIA LOS	0	1	0	1	2	3	4	5	6	7	0	1	2	3	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7		
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		
USB 3.2 Lanes	1	2																													2	
PCIe - Gen4 Lanes			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20									9	8
PCIe - Gen5 Lanes																							21	22	23	24	25	26	27	28		
GbE Lanes							X																								0	1
SATA Lanes			0	1																											2	

- FIA = Flex-IO Adapter
- FIA LOS = Flex-IO Adapter Lane Ownership Number

The 30 Flexible HSIO Lanes [29:0] support the following:

1. Up to twenty eight PCIe* Lanes
   - A maximum of nine PCIe* Root Ports (or devices) can be enabled when GbE Port is disabled
   - A maximum of eight PCIe* Root Ports (or devices) can be enabled when GbE Port is enabled
   - PCIe* Lanes 1-4 (PCIe* Controller #1 Gen4), 5-8 (PCIe* Controller #2 Gen4), 9-12 (PCIe* Controller #3 Gen4), 13-16 (PCIe* Controller #4 Gen4), 17-20 (PCIe* Controller #5 Gen4), and 21-28 (PCIe* Controller #6 Gen5) must be individually configured
2. Up to two USB 3.2 Gen 1x1/2x1 Lanes

- A maximum of two USB 3.2 Gen 1x1/2x1 Ports (or devices) can be enabled
- USB 3.2 Gen 1x1 = First Generation with One 5 GT/s Data Lane
- USB 3.2 Gen 2x1 = Second Generation with One 10 GT/s Data Lane

3. Up to two SATA Lanes
   - A maximum of two SATA Ports (or devices) can be enabled
4. Up to one GbE Lane
   - A maximum of one GbE Port can be enabled

1.11 Related Documents

Document	Document Number
Intel® Core™ Ultra 200H and 200U Series Processors Datasheet Volume 2 of 2	844261

2.0 Processor and Device IDs

2.1 CPUID

Table 5. CPUID Format

SKU	CPUID	Reserved [31:28]	Extended Family [27:20]	Extended Model [19:16]	Reserved [15:14]	Processor Type [13:12]	Family Code [11:8]	Model Number [7:4]	Stepping ID [3:0]
H 6P+8E	C0652h	Reserved	0000000b	1100b	Reserved	00b	0110b	0101b	0010b
U 2P+8E	B0650h	Reserved	0000000b	1011b	Reserved	00b	0110b	0101b	0000b

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

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

2.2 PCI Configuration Header

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

Table 6. PCI Configuration Header

Byte3	Byte2	Byte1	Byte0	Address
Device ID		**Vendor ID (8086h)**		00h
Status		Command		04h
Class Code			**Revision ID**	08h
BIST	Header Type	Latency Timer	Cache Line Size	0Ch
Base Address Register0 (BAR0)				10h
Base Address Register1 (BAR1)				14h
Base Address Register2 (BAR2)				18h
Base Address Register3 (BAR3)				1Ch
Base Address Register4 (BAR4)				20h
Base Address Register5 (BAR5)				24h
Card-bus CIS Pointer				28h
Subsystem ID		**Subsystem Vendor ID**		2Ch
Expansion ROM Base Address				30h
Reserved			Capabilities Pointer	34h
Reserved				38h
Maximum Latency	Minimum Grant	Interrupt Pin	Interrupt Line	3Ch

2.3 Device IDs

This section specifies the device IDs of the processor.

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

Processor Line	Package	Compute Tile P-Cores	Compute Tile E-Cores	SOC Tile E-cores	Graphics Configuration Xe-cores	Host Device ID (DID0)	Processor Graphics Device ID (DID2)
H 6P+8E	BGA2049	6	8	2	8	7D06h	7D51h
U 2P+8E	BGA2049	2	8	2	8	7D30h	7D41h

Table 8. Other Device ID

Device	Bus / Device / Function	H/U Processor Series DID
PCI Express* Root Port #12 (H-Processor Series PEG) (PXPF)	0 / 1 / 0	7ECCh
Dynamic Tuning Technology (DTT)	0 / 4 / 0	7D03h
IPU	0 / 5 / 0	7D19h
PCI Express Root Port #9 (PXPC)	0 / 6 / 0	774Dh
PCI Express Root Port #10 (PXPD)	0 / 6 / 1	7ECAh
PCI Express Root Port #11 (PXPE)	0 / 6 / 2	7ECBh
USB Type-C* Subsystem PCIe Root Port #16	0 / 7 / 0	7EC4h

continued...

Device	Bus / Device / Function	H/U Processor Series DID
USB Type-C Subsystem PCIe Root Port #17	0 / 7 / 1	7EC5h
USB Type-C Subsystem PCIe Root Port #18	0 / 7 / 2	7EC6h
USB Type-C Subsystem PCIe Root Port #19	0 / 7 / 3	7EC7h
Gaussian & Neural-Network Accelerator (GNA)	0 / 8 / 0	774Ch
Platform Monitoring Technology (PMT)	0 / 10 / 0	7D0Dh
NPU	0 / 11 / 0	7D1Dh
USB xHCI	0 / 13 / 0	7EC0h
USB xDCI	0 / 13 / 1	7EC1h
Thunderbolt™ DMA0	0 / 13 / 2	7EC2h
Thunderbolt™ DMA1	0 / 13 / 3	7EC3h
Intel® Volume Management Device (VMD)	0 / 14 / 0	7D0Bh
THC #0 (Touch Host Controller) ID1	0 / 16 / 0	7748h
THC #0 (Touch Host Controller) ID2	0 / 16 / 0	7749h
THC #1 (Touch Host Controller) ID1	0 / 16 / 1	774Ah
THC #1 (Touch Host Controller) ID2	0 / 16 / 1	774Bh
Integrated Sensor Hub	0 / 18 / 0	7745h
GSPI #2	0 / 18 / 6	7746h
P2SB (IOE)	0 / 19 / 0	7EC8h
IEH (IOE)	0 / 19 / 1	7EC9h
PMC (IOE)	0 / 19 / 2	7ECEh
Shared SRAM (IOE)	0 / 19 / 3	7ECFh
Standalone xHCI Controller	0 / 20 / 0	777Dh
Standalone USB Device Controller	0 / 20 / 1	777Eh
Shared SRAM	0 / 20 / 2	777Fh
CNVi: Wi-Fi*	0 / 20 / 3	7740h
I^2C Controller #0	0 / 21 / 0	7778h
I^2C Controller #1	0 / 21 / 1	7779h
I^2C Controller #2	0 / 21 / 2	777Ah
I^2C Controller #3	0 / 21 / 3	777Bh
I^3C Controller	0 / 21 / 4	777Ch
Intel® CSME: HECI #1	0 / 22 / 0	7770h
Intel® CSME: HECI #2	0 / 22 / 1	7771h
Intel® CSME: IDE Redirection (IDE-R)	0 / 22 / 2	7772h
Intel® CSME: Keyboard and Text (KT) Redirection	0 / 22 / 3	7773h
Intel® CSME: HECI #3	0 / 22 / 4	7774h

continued...

Device	Bus / Device / Function	H/U Processor Series DID
Intel® CSME: HECI #4	0 / 22 / 5	7775h
SATA Controller (AHCI)	0 / 23 / 0	7763h
SATA Controller (RAID 0/1/5/10) - premium	0 / 23 / 0	7767h
SATA Controller (RAID 0/1/5/10) - Inbox Compatible ID	0 / 23 / 0	282Ah
Intel® CSME: HECI #1	0 / 24 / 0	7758h
Intel® CSME: HECI #2	0 / 24 / 1	7759h
Intel® CSME: HECI #3	0 / 24 / 2	775Ah
I^2C Controller #4	0 / 25 / 0	7750h
I^2C Controller #5	0 / 25 / 1	7751h
UART #2	0 / 25 / 2	7752h
PCI Express Root Port #1	0 / 28 / 0	7738h
PCI Express Root Port #2	0 / 28 / 1	7739h
PCI Express Root Port #3	0 / 28 / 2	773Ah
PCI Express Root Port #4	0 / 28 / 3	773Bh
PCI Express Root Port #5	0 / 28 / 4	773Ch
PCI Express Root Port #6	0 / 28 / 5	773Dh
PCI Express Root Port #7	0 / 28 / 6	773Eh
PCI Express Root Port #8	0 / 28 / 7	773Fh
UART #0	0 / 30 / 0	7725h
UART #1	0 / 30 / 1	7726h
GSPI #0	0 / 30 / 2	7727h
GSPI #1	0 / 30 / 3	7730h
eSPI Controller	0 / 31 / 0	H: 7202h U: 7203h
P2SB (SOC)	0 / 31 / 1	7720h
PMC (SOC)	0 / 31 / 2	7721h
Intel® High Definition Audio (Intel® HD Audio) AVS (Audio, Voice, Speech)	0 / 31 / 3	7728h
SMBus	0 / 31 / 4	7722h
SPI (flash) Controller	0 / 31 / 5	7723h
GbE Controller: Corporate/Intel® vPro™ (Default)	0 / 31 / 6	57A0h
GbE Controller: Consumer	0 / 31 / 6	57A1h
Intel® Trace Hub (Intel® TH)	0 / 31 / 7	7724h

Table 9. ACPI Device ID for GPIO Controller

ACPI ID	Note
H and U-Processor Series:	INTC105E

2.4 Revision IDs

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

3.0 Package Mechanical Specifications

3.1 Package Mechanical Attributes

The U/H-series processor use a Flip Chip technology available in a Ball Grid Array (BGA) package. The following table provides an overview of the package mechanical attributes.

Table 10. H/U Package Mechanical Attributes

Package	Parameter	H/U-Series Processor
Package Technology	Package Type	Flip Chip Ball Grid Array
	Interconnect	Ball Grid Array (BGA)
	Lead Free	Yes
	Halogenated Flame Retardant Free	Yes
Package Configuration	Solder Ball Composition	SAC405
	Ball/Pin Count	2049
	Grid Array Pattern	Balls anywhere
	Land Side Capacitors	Yes
	Tile Side Capacitors	No
	Tile Configuration	Foveros
Package Dimensions	Nominal Package Size	25 x 50 mm
	Z	Substrate Z = 0.644 ± 0.095 mm 1.245 ± 0.109 mm (BOTTOM OF BGA TO TOP OF TILE)
	Minimum Ball/Pin pitch	0.65 mm BP

3.2 Package Loading and Tile Pressure Specifications

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

3.2.1 Static Compressive Load Specification

- Considerations should be made to ensure steady state static loading on the packages that does not exceed the limits recommended. Excessive steady state static loading can induce solder ball cracks, especially over a period of time resulting in higher failure rate.
- This static compressive load is not to be exceeded; therefore the tolerance of the package and the tolerances of the thermal solution (including attach mechanism) should be taken into account when calculating or measuring static load on the package.

- An ideal thermal solution design would apply a load as uniform as possible on all dies in order to optimize thermal performance and minimize mechanical risk

<table>
<tr><th rowspan="2">Package</th><th rowspan="2">PCB Thickness</th><th rowspan="2">Back Plate Configuration</th><th colspan="4">Supported Load</th></tr>
<tr><th><=10 lbf</th><th>10.1 lbf - 15 lbf</th><th>15.1 lbf - 20 lbf</th><th>20.1 lbf - 25 lbf</th></tr>
<tr><td rowspan="4">U/H</td><td rowspan="2">0.70mm -0.79mm / 28mil -31.9mil</td><td>No Backplate</td><td rowspan="2">Corner Glue Allowed</td><td>Corner Glue Allowed</td><td>Not Supported</td><td rowspan="2">Not Supported</td></tr>
<tr><td>Back Plate [3]</td><td>Corner Glue Required [2]</td><td>Corner Glue Required [2]</td></tr>
<tr><td rowspan="2">0.80mm - 1.2mm / 32mil -48mil</td><td>No Backplate</td><td colspan="2" rowspan="2">Corner Glue Allowed</td><td colspan="2">Not Supported</td></tr>
<tr><td>Back Plate [3]</td><td colspan="2">Corner Glue Required [2]</td></tr>
<tr><td colspan="7">Notes: 1. If using backplate, recommended maximum back plate thickness is 0.5mm.
2. This specification is based on limited testing for design characterization.
3. All values are pre-silicon values and are subject to be changed.</td></tr>
</table>

3.2.2 Maximum Pressure Specifications

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

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

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

Measurement Method: Intel has provided the document for accurate measurement of pressure on tiles.

Table 11. Package Loading Specifications

Package	Static Compressive Pressure[1] [PSI]	Transient Compressive Pressure[1] [PSI]
U/H	800	800
Note: This is the load and pressure that has been tested by Intel for a single assembly cycle. This metric is a pressure over 2 mm^2 (2 mm x 2 mm) area.		

3.3 Package Storage Specifications

Parameter	Description	Minimum	Maximum
$T_{ABSOLUTE\ STORAGE}$	The non-operating device storage temperature. Damage (latent or otherwise) may occur when subjected to this temperature for any length of time in Intel Original sealed moisture barrier bag and / or box.	- 25°C	125°C
$T_{SUSTAINED\ STORAGE}$	The ambient storage temperature limit (in shipping media) for the sustained period of time	-5°C	40°C
$RH_{SUSTAINED\ STORAGE}$	The maximum device storage relative humidity for the sustained period of time as specified below in Intel Original sealed moisture barrier bag and / or box	60% @ 24°C	
$TIME_{SUSTAINED\ STORAGE}$	Maximum time: associated with customer shelf life in Intel Original sealed moisture barrier bag and / or box	NA	**Moisture Sensitive Devices:** 60 months from bag seal date; **Non-moisture sensitive devices:** 60 months from lot date
Storage Conditions	Processors in a non-operational state may be installed in a platform, in a tray, boxed, or loose and may be sealed in airtight package or exposed to free air. Under these conditions, processor landings should not be connected to any supply voltages, have any I/Os biased, or receive any clocks. Upon exposure to "free air" (that is, unsealed packaging or a device removed from packaging material), the processor should be handled in accordance with moisture sensitivity labeling (MSL) as indicated on the packaging material. Boxed Land Grid Array packaged (LGA) processors are MSL 1 ('unlimited' or unaffected) as they are not heated in order to be inserted in the socket.		

Notes:
1. $T_{ABSOLUTE\ STORAGE}$ applies to the un-assembled component only and does not apply to the shipping media, moisture barrier bags or desiccant. Refers to a component device that is not assembled in a board or socket that is not to be electrically connected to a voltage reference or I/O signals.
2. Specified temperatures are based on data collected. Exceptions for surface mount re-flow are specified by applicable JEDEC J-STD-020 and MAS documents. The JEDEC, J-STD-020 moisture level rating and associated handling practices apply to all moisture sensitive devices removed from the moisture barrier bag.
3. Post board attaches storage temperature limits are not specified for non-Intel branded boards. Consult your board manufacturer for storage specifications.

4.0 Memory Mapping

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

4.1 Functional Description

4.1.1 PCI Devices and Functions

The Processor incorporates a variety of PCI devices and functions, as shown in the following table. If for some reason, the particular system platform does not want to support any one of the Device Functions, with the exception of D30:F0, they can individually be disabled. The integrated Gigabit Ethernet controller will be disabled if no Platform LAN Connect component is detected (Gigabit Ethernet Controller on page 238). When a function is disabled, it does not appear to the software. A disabled function will not respond to any register reads or writes, ensuring that these devices appear hidden to software.

4.1.2 Fixed I/O Address Ranges

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

NOTE

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

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

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

Table 12. Fixed I/O Ranges Decoded by Processor

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

continued...

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

continued...

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

continued...

I/O Address	Read Target	Write Target	Internal Unit (Unless[E]: External)	Separate Enable/Disable
200-207h	Gameport Low	Gameport Low	Forwarded to eSPI	Yes. ESPI_CS1IORE.LGE
208-20Fh	Gameport High	Gameport High	Forwarded to eSPI	Yes ESPI_CS1IORE.HGRE
4D0h – 4D1h	Interrupt Controller	Interrupt Controller	Interrupt Controller	None
CF9h	Reset Generator	Reset Generator	Interrupt controller	None
Notes: 1. Only if the Port 61 Alias Enable bit (GIC.P61AE) bit is set. Otherwise, the cycle is internally terminated by the Processor. 2. This includes byte, word, or double-word (DW) access at I/O address 80h.				

4.1.3 Variable I/O Decode Ranges

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

WARNING

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

Table 13. Variable I/O Decode Ranges

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

continued...

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

4.2 Memory Map

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

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

NOTE

Total ports are different for the different SKUs.

Table 14. Processor Memory Decode Ranges (Processor Perspective)

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

continued...

Memory Range	Target	Dependency/Comments
FEC8 0000 - FEC8 7FFF	PCIe port 15	PCIe root port 15 I/OxApic Enable (PAE) is set
FEC8 8000 - FEC8 FFFF	PCIe port 16	PCIe root port 16 I/OxApic Enable (PAE) is set
FEC9 0000 - FEC9 7FFF	PCIe port 17	PCIe root port 17 I/OxApic Enable (PAE) is set
FEC9 8000 - FEC9 FFFF	PCIe port 18	PCIe root port 18 I/OxApic Enable (PAE) is set
FECA 0000 - FECA 7FFF	PCIe port 19	PCIe root port 19 I/OxApic Enable (PAE) is set
FECA 8000 - FECA FFFF	PCIe port 20	PCIe root port 20 I/OxApic Enable (PAE) is set
FECB 0000 - FECB 7FFF	PCIe port 21	PCIe root port 21 I/OxApic Enable (PAE) is set
FECB 8000 - FECB FFFF	PCIe port 22	PCIe root port 22 I/OxApic Enable (PAE) is set
FECC 0000 - FECC 7FFF	PCIe port 23	PCIe root port 23 I/OxApic Enable (PAE) is set
FECC 8000 - FECC FFFF	PCIe port 24	PCIe root port 24 I/OxApic Enable (PAE) is set
FECD 0000 - FECD 7FFF	PCIe port 25	PCIe root port 25 I/OxApic Enable (PAE) is set
FECD 8000 - FECD FFFF	PCIe port 26	PCIe root port 26 I/OxApic Enable (PAE) is set
FECE 0000 - FECE_7FFF	PCIe port 27	PCIe root port 27 I/OxApic Enable (PAE) is set
FECE 8000 - FECE FFFF	PCIe port 28	PCIe root port 28 I/OxApic Enable (PAE) is set
FEF0 0000 - FEFF FFFF	eSPI or SPI	uCode Patch Region Enable UCPR.UPRE is set
FFC0 0000 - FFC7 FFFF FF80 0000 - FF87 FFFF	eSPI or SPI	Bit 8 in BIOS Decode Enable Register is set
FFC8 0000 – FFCF FFFF FF88 0000 - FF8F FFFF	eSPI or SPI	Bit 9 in BIOS Decode Enable Register is set
FFD0 0000 - FFD7 FFFF FF90 0000 - FF97 FFFF	eSPI or SPI	Bit 10 in BIOS Decode Enable Register is set
FFD8 0000 – FFDF FFFF FF98 0000 - FF9F FFFF	eSPI or SPI	Bit 11 in BIOS Decode Enable Register is set
FFE0 0000 - FFE7 FFFF FFA0 0000 - FFA7 FFFF	eSPI or SPI	Bit 12 in BIOS Decode Enable Register is set
FFE8 0000 – FFEF FFFF FFA8 0000 – FFAF FFFF	eSPI or SPI	Bit 13 in BIOS Decode Enable Register is set
FFF0 0000 - FFF7 FFFF FFB0 0000 - FFB7 FFFF	eSPI or SPI	Bit 14 in BIOS Decode Enable Register is set
FFFC 0000 - FFFF FFFF	eSPI, SPI, or Intel® CSME	Always enabled. Refer to Table 15 on page 45 for swappable ranges
FFF8 0000 - FFFB FFFF FFB8 0000 - FFBF FFFF	eSPI or SPI	Always enabled. Refer to Table 15 on page 45 for swappable ranges
FF70 0000 - FF7F FFFF FF30 0000 - FF3F FFFF	eSPI or SPI	Bit 3 in BIOS Decode Enable Register is set
FF60 0000 - FF6F FFFF FF20 0000 - FF2F FFFF	eSPI or SPI	Bit 2 in BIOS Decode Enable Register is set
FF50 0000 - FF5F FFFF FF10 0000 - FF1F FFFF	eSPI or SPI	Bit 1 in BIOS Decode Enable Register is set
FF40 0000 - FF4F FFFF	eSPI or SPI	Bit 0 in BIOS Decode Enable Register is set

continued...

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

continued...

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

4.2.1 Boot Block Update Scheme

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

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

Table 15. Boot Block Update Scheme

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

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

The algorithm is:

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

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

5.0 Security Technologies

5.1 Intel® Converged Boot Guard and Intel® TXT

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

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

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

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

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

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

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

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

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

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

5.2 Crypto Acceleration Instructions

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

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

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

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

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

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

NOTE

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

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

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

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

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

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

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

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

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

5.2.4 New Cryptographic Acceleration Instructions

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

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

5.3 Intel® Secure Key

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

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

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

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

5.4 Execute Disable Bit

The Execute Disable Bit allows memory to be marked as non-executable when combined with a supporting operating system. If code attempts to run in non-executable memory, the processor raises an error to the operating system. This feature can prevent some classes of viruses or worms that exploit buffer overrun vulnerabilities and can, thus, help improve the overall security of the system.

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

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

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

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

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

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

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

5.7 User Mode Instruction Prevention (UMIP)

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

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

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

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

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

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

5.8 Read Processor ID (RDPID)

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

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

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

5.9 Intel® Total Memory Encryption - Multi-Key

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

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

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

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

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

NOTES

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

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

5.10 Control-flow Enforcement Technology (Intel® CET)

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

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

5.10.1 Shadow Stack

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

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

5.10.2 Indirect Branch Tracking

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

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

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

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

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

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

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

The Platform Properties Assessment Module (PPAM) is the primary component of Intel® System Security Report. It collects and reports information about platform SMM implementation and configuration, in order to provide trustworthy attestation of the

resulting SMI memory policy regarding SMM secure configuration and access to MLE owned memory. Intel® System Security Report is used to create a trustworthy report describing the SMM policy. PPAM is a major/core component of Intel® System Security Report 1.0/1.1 technology

5.12 BIOS Guard

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

5.13 Intel® Platform Trust Technology

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

5.14 Linear Address Space Separation (LASS)

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

5.15 Security Firmware Engines

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

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

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

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

5.15.2 Intel® Silicon Security Engine

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

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

5.15.3 Intel® Graphics System Controller (Intel® GSC)

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

NOTE

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

6.0 Intel® Virtualization Technology (Intel® VT)

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

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

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

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

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

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

6.1 Intel® Virtualization Technology (Intel® VT) for Intel® 64 and Intel® Architecture (Intel® VT-x)

Intel® VT-x Objectives

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

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

Intel® VT-x Key Features

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

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

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

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

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

- **EPTP (EPT pointer) switching**

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

- **Pause loop exiting**

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

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

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

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

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

- **Virtualization Exceptions**

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

- **Translation of Guest-Physical Addresses Used by Intel Processor Trace**

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

6.2 Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d)

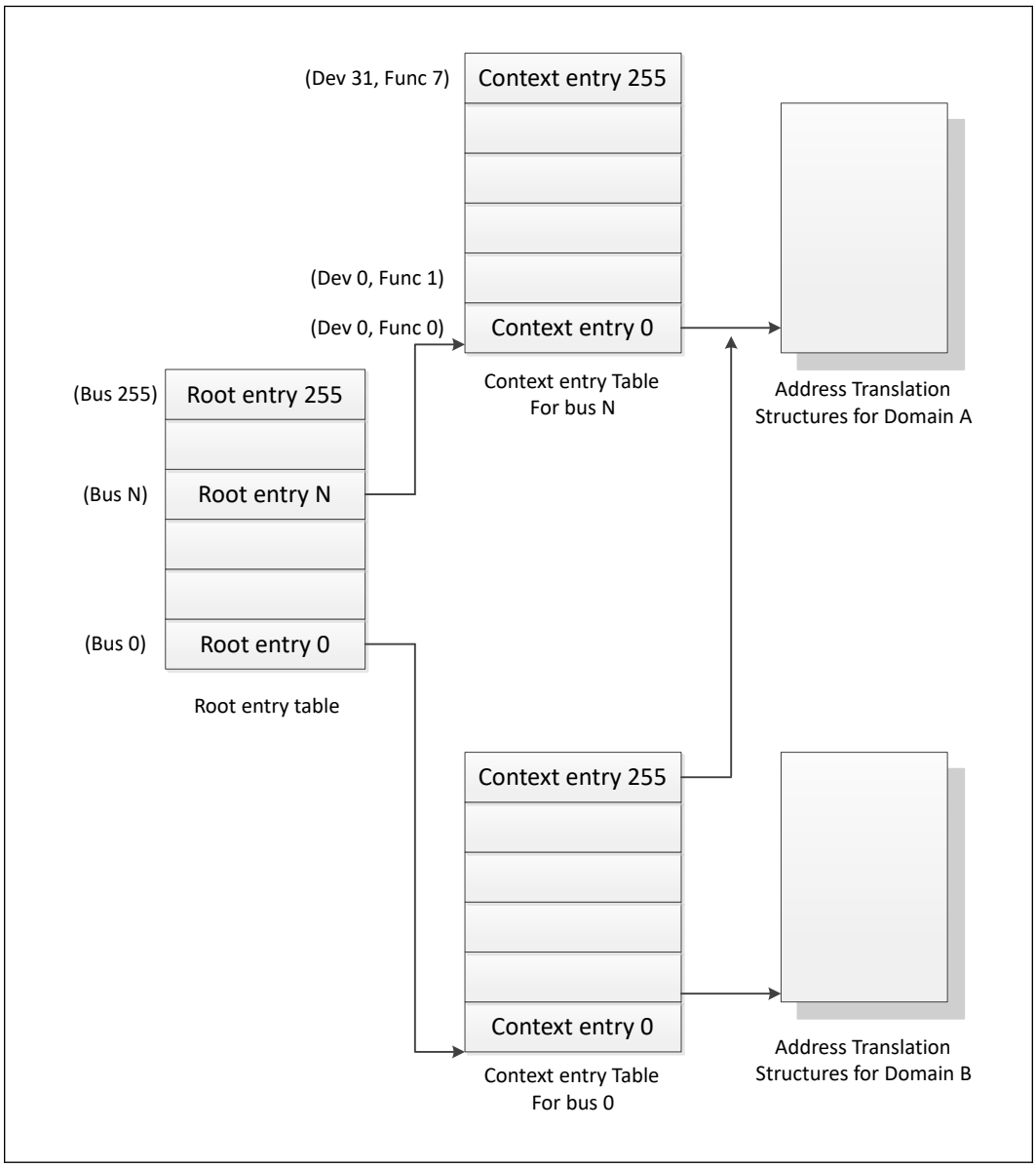
Intel® VT-d Objectives

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

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

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

Figure 4. Device to Domain Mapping Structures



Intel® VT-d functionality often referred to as an Intel® VT-d Engine, has typically been implemented at or near a PCI Express* host bridge component of a computer system. This might be in the PCI Express functionality of a processor with integrated I/O. When one such VT-d engine receives a PCI Express transaction from a PCI Express bus, it uses the B/D/F number associated with the transaction to search for an Intel® VT-d translation table. In doing so, it uses the B/D/F number to traverse the data structure shown in the above figure. If it finds a valid Intel® VT-d table in this data structure, it uses that table to translate the address provided on the PCI Express bus. If it does not find a valid translation table for a given translation, this results in an Intel® VT-d fault. If Intel® VT-d translation is required, the Intel® VT-d engine performs an N-level table walk.

For more information, refer to *Intel® Virtualization Technology for Directed I/O Architecture Specification* http://www.intel.com/content/dam/www/public/us/en/documents/product-specifications/vt-directed-io-spec.pdf

Intel® VT-d Key Features

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

- Memory controller and processor graphics comply with the Intel® VT-d 2.1 Specification.
- Two Intel® VT-d DMA remap engines.
  - iGFX DMA remap engine
  - Default DMA remap engine (covers all devices except iGFX)
- 46-bit guest physical address and host physical address widths
- Support for register-based fault recording only (for single entry only) and support for MSI interrupts for faults
- Support for both leaf and non-leaf caching
- Support for non-caching of invalid page table entries
- Support for hardware-based flushing of translated but pending writes and pending reads, on IOTLB invalidation
- Support for Global, Domain-specific and Page specific IOTLB invalidation
- Interrupt Remapping is supported
- Queued invalidation is supported
- 4-level Intel®VT-d Page walk - all VTd engines support 4-level tables only (adjusted guest address width of 48 bits)
- Intel®VT-d super-page - all VTd engines support super-page (2 MB, 1 GB)
- Scalable Mode - all VTd engines support Scalable mode operation (using RID_PASID only)
- Nested - default Intel ® VT-d engine support Nested translation

NOTE

Intel® VT-d Technology may not be available on all SKUs.

6.3 Intel® APIC Virtualization Technology (Intel® APICv)

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

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

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

- **Virtual-interrupt Delivery:** This controls enables the evaluation and delivery of pending virtual interrupts. It also enables the emulation of writes (memory-mapped or MSR-based, as enabled) to the APIC registers that control interrupt prioritization.
- **Use TPR Shadow:** This control enables emulation of accesses to the APIC's task-priority register (TPR) via CR8 and, if enabled, via the memory-mapped or MSR-based interfaces.
- **Virtualize APIC Accesses:** This control enables virtualization of memory-mapped accesses to the APIC by causing VM exits on accesses to a VMM-specified APIC-access page. Some of the other controls, if set, may cause some of these accesses to be emulated rather than causing VM exits.
- **Virtualize x2APIC Mode:** This control enables virtualization of MSR-based accesses to the APIC.
- **APIC-register Virtualization:** This control allows memory-mapped and MSR-based reads of most APIC registers (as enabled) by satisfying them from the virtual-APIC page. It directs memory-mapped writes to the APIC-access page to the virtual-APIC page, following them by VM exits for VMM emulation.
- **Process Posted Interrupts:** This control allows software to post virtual interrupts in a data structure and send a notification to another logical processor; upon receipt of the notification, the target processor will process the posted interrupts by copying them into the virtual-APIC page.

NOTE

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

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

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

7.0 Platform Environmental Control Interface (PECI)

PECI is an Intel proprietary interface that provides a communication channel between Intel processors and external components such as Super IO (SIO) and Embedded Controllers (EC) to provide processor temperature, Turbo, Assured Power (cTDP), and Memory Throttling Control mechanisms and many other services. PECI is used for platform thermal management and real-time control and configuration of processor features and performance.

NOTE

- PECI over eSPI is supported.

7.1 PECI Bus Architecture

The PECI architecture is based on a wired-OR bus that the clients (as processor PECI) can pull up (with the strong drive).

The idle state on the bus is ‘0’ (logical low) and near zero (Logical voltage level).

NOTE

PECI supported frequency range is 100 Khz-1 MHz.

The following figures demonstrate PECI design and connectivity:

- PECI Host-Clients Connection: While the host/originator can be third party PECI host and one of the PECI clients is a processor PECI device.
- PECI EC Connection.

Figure 5. PECI Host-Clients Connection Example

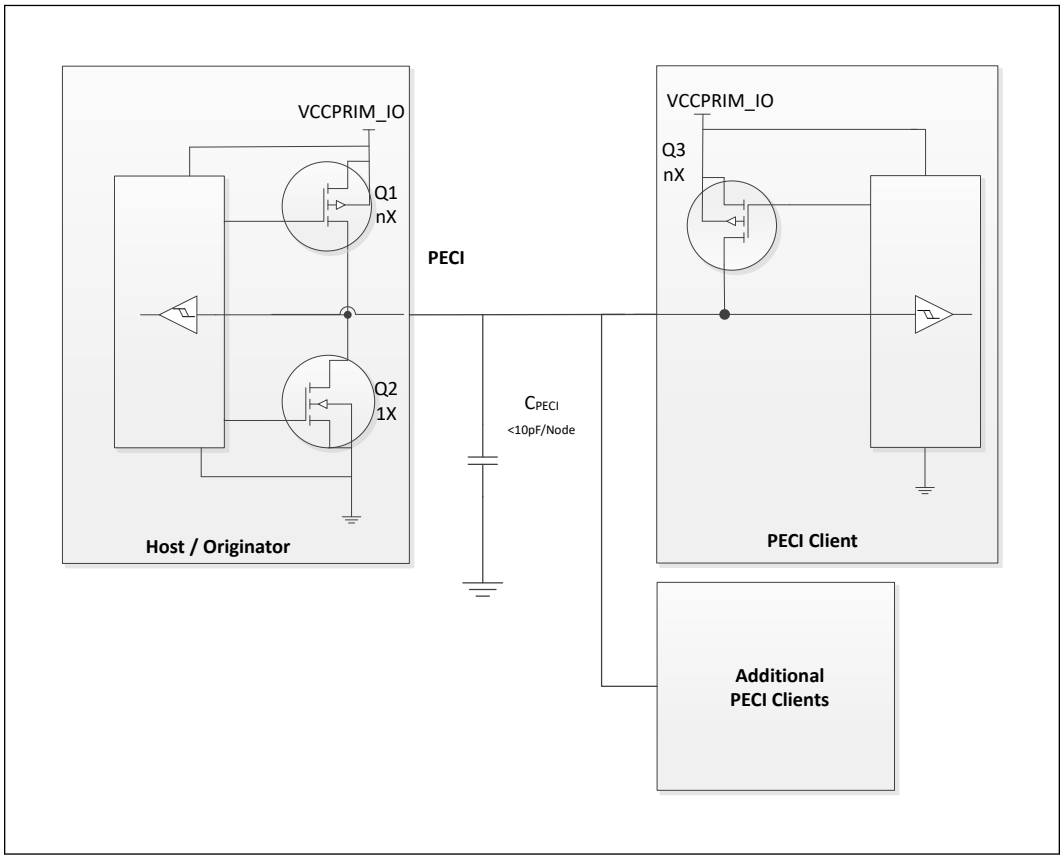
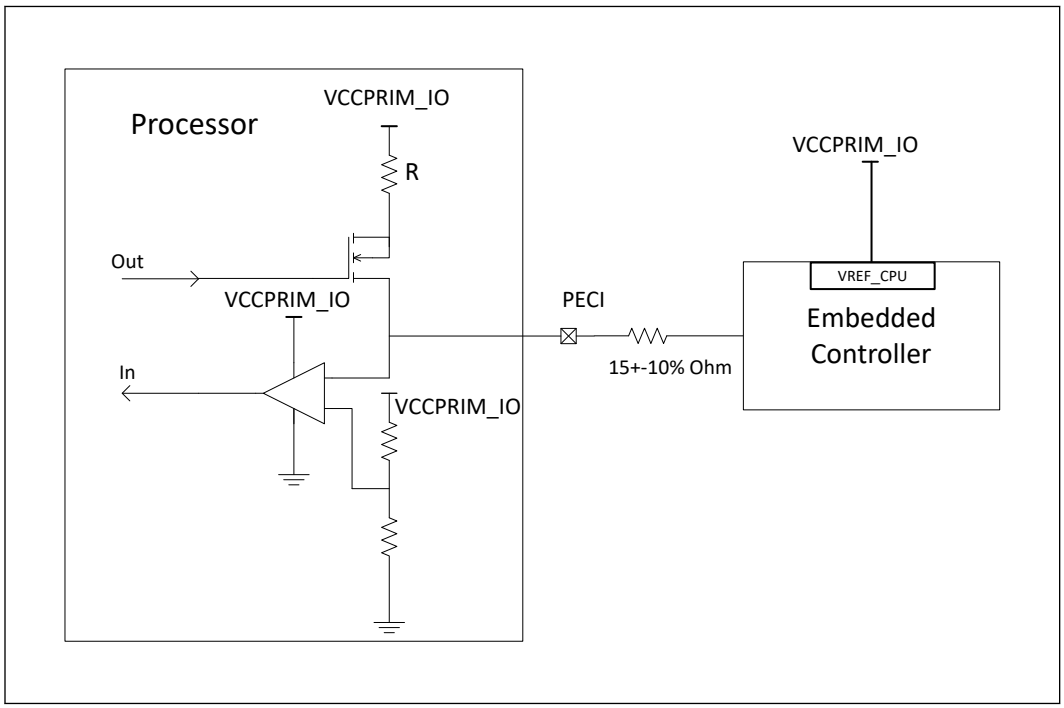


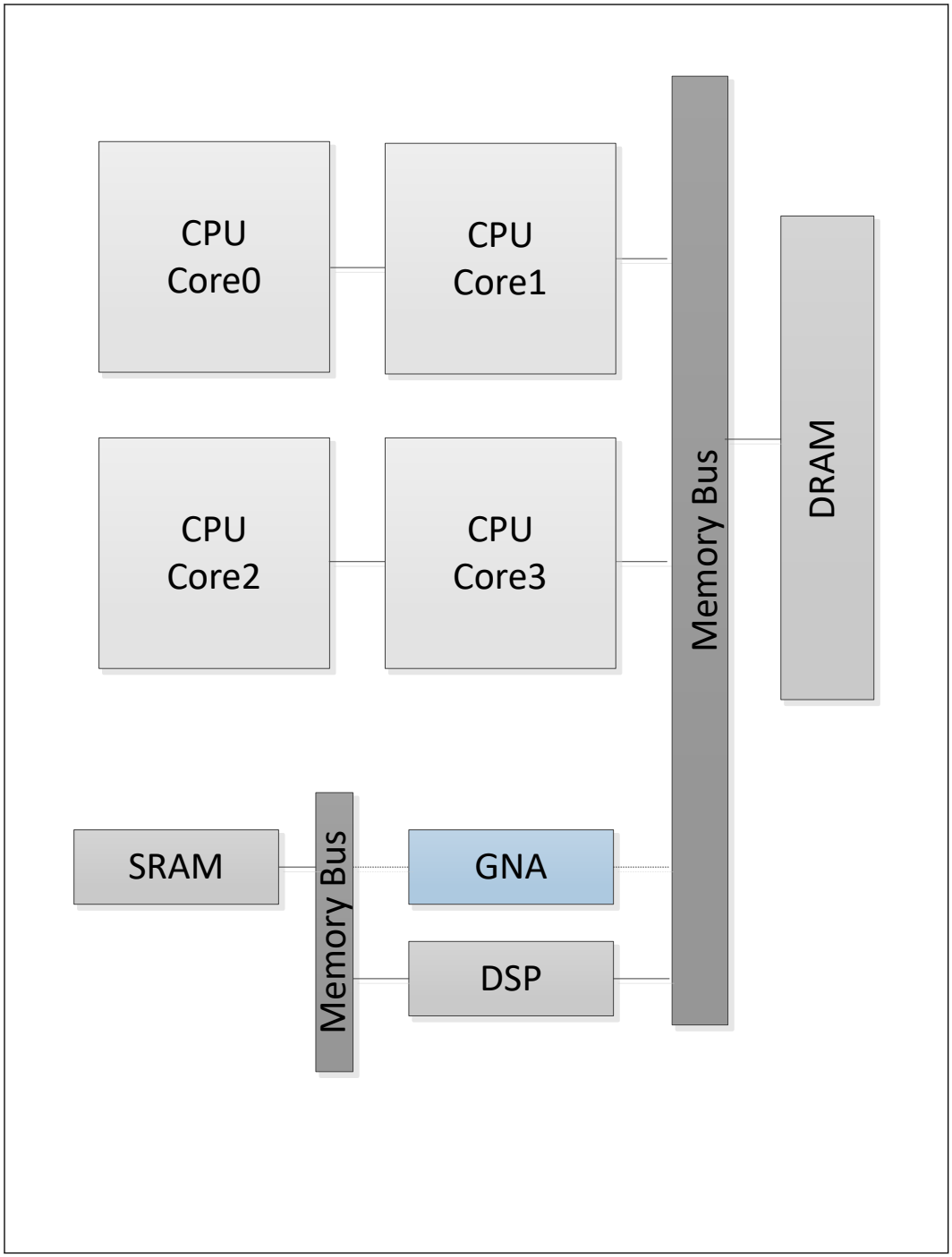
Figure 6. PECI EC Connection Example



8.0 Intel® GMM and Neural Network Accelerator (Intel® GNA 3.0)

GNA stands for Gaussian Mixture Model and Neural Network Accelerator.

The GNA is used to process speech recognition without user training sequence. The GNA is designed to unload the processor cores and the system memory with complex speech recognition tasks and improve the speech recognition accuracy. The GNA is designed to compute millions of Gaussian probability density functions per second without loading the processor cores while maintaining low power consumption.


CPU
Core0
CPU
Core1
CPU
Core2
CPU
Core3
Memory Bus
DRAM
SRAM
Memory Bus
GNA
DSP

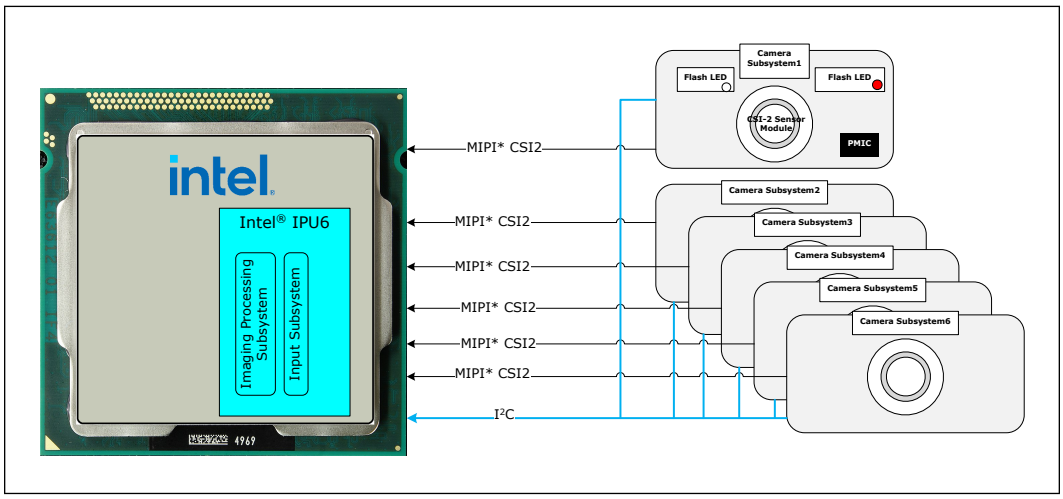
9.0 Intel® Image Processing Unit (Intel® IPU6)

9.1 Platform Imaging Infrastructure

The platform imaging infrastructure is based on the following hardware components:

- **Camera Subsystem:** Located in the lid of the system and contains CMOS sensor, flash, LED, I/O interface (MIPI* CSI-2 and I2C*), focus control and other components.
- **Camera I/O Controller:** Located in the processor and contains I2C controller devices for camera control and GPIO controllers.
- **Intel® IPU (Image Processing Unit):** The IPU includes two main components under the same PCIe device - The first component is an input system which includes PHYs and CSI-2 controllers to capture MIPI data and convert it to pixel data from multiple cameras. The second component is a processing system which processes raw Bayer format using a high quality and performance HW image processing pipeline. The result images are used by still photography and video capture applications (JPEG, H.264, and so on.).

Figure 7. Processor Camera System



NOTE

This diagram is general. For specific Intel® Core™ Ultra Processor configuration, refer to MIPI* CSI-2 Camera Interconnect on page 67.

9.2 Intel® Image Processing Unit (Intel® IPU6)

IPU6 is Intel's sixth generation solution for an Imaging Processing Unit, providing advanced imaging functionality for Intel® Core™ branded processors, Tablets, Automotive, Digital Surveillance Systems (DSS), and other market segments.

IPU6 is a continuing evolution of the architecture introduced in IPU4 and enhanced in IPU5. Additional image quality improvements are introduced, as well as hardware accelerated support for temporal de-noising and new sensor technologies such as Spatially Variant Exposure HDR and Dual Photo Diode, among others.

IPU6 provides a complete high-quality hardware accelerated pipeline.

9.3 Camera/MIPI

Camera/MIPI is supported on the H/U-series processor.

9.3.1 Camera Pipe Support

The IPU6EP fixed function pipe supports the following functions:

- Black level compensation (BLC)
- White balance
- Color matching
- Lens shading (vignette) correction (LSC)
- Defect pixel correction (DFC)
- Color crosstalk (color shading) correction
- Dynamic defect pixel replacement
- Auto-focus-pixel (PDAF) hiding
- High quality demosaicing
- Scaling and format conversions with arbitrary aspect ratios
- Spatiotemporal noise suppression
- Sensor types: RGB Bayer, IR (mono), RGB-IR 4x4 hybrid.
- Actuator types: Voice coil AF, PDAF T1-T3, both in video and still modes
- LED flash
- Internal face detection utilized in 3A statistics (AWB, AE, AF and gamma)
- Concurrent processing of camera streams with time-multiplexing

  (Limited by processing bandwidth and number of cameras ports physically connected)
- Integration with 3rd party Computer Vision solutions.

9.3.2 MIPI* CSI-2 Camera Interconnect

The Camera I/O Controller provides a native/integrated interconnect to camera sensors, compliant with MIPI* CSI-2 V2.0 protocol. Total of 8 data + 4 clock lanes (H/U) are available for the camera interface supporting up to 4 sensors connected with 3 concurrent in operation.

Data transmission interface (referred as CSI-2) is a unidirectional differential serial interface with data and clock signals; the physical layer of this interface is the MIPI* Alliance Specification for D-PHY.

The control interface (referred as CCI) is a bi-directional control interface compatible with I^2C standard.

9.3.2.1 Camera Control Logic

The camera infrastructure supports several architectural options for camera control utilizing camera PMIC and/or discrete logic. IPU6 control options utilize I^2C for bidirectional communication and GPIOs to drive various control functions.

9.3.2.2 Camera Modules

Intel maintains an Intel User Facing (UF) and Infra-red (IR) Camera Approved Vendor List and Intel World Facing (WF) Approved Vendor List to simplify system design. Additional services are available to support non-AVL options.

9.3.2.3 MIPI* CSI-2 Interface Signals

Signal Name	Design Pin Name	Description	Dir.	Buffer Type	Link Type	H/U-Series Processor
CSI_A_DP[3:0] CSI_A_DN[3:0]	CSI_A_DN[0] CSI_A_DP[0] CSI_A_DN[1] CSI_A_DP[1] *Note:* Next pins assigned under port B	CSI-2 Port A Data lane	I	DPHY	Diff	Supports X1/X2/X4 *Note:* Upper lanes CSI_A[3:2] available only when CSI_B[1:0] are not used as a separate camera.
CSI_B_DP[1:0] CSI_B_DN[1:0]	CSI_B_DN[0]/ CSI_A_DN[2] CSI_B_DP[0]/ CSI_A_DP[2] CSI_B_DN[1]/ CSI_A_DN[3] CSI_B_DP[1]/ CSI_A_DP[3]	CSI-2 Port B Data lane **OR** Continuation of CSI-2 Port A Data lane	I	DPHY	Diff	Supports X1/X2 *Note:* If CSI_B[1:0] are used as a separate camera CSI_A[3:2] are not supported.
CSI_E_DP[3:0] CSI_E_DN[3:0]	CSI_E_DN[0] CSI_E_DP[0] CSI_E_DN[1] CSI_E_DP[1] *Note:* Next pins assigned under port F	CSI-2 Port E Data lane	I	DPHY	Diff	Supports X1/X2/X4 *Note:* Upper lanes CSI_E[3:2] available only when CSI_F[1:0] are not used as a separate camera.
CSI_F_DP[1:0] CSI_F_DN[1:0]	CSI_F_DN[0]/ CSI_E_DN[2] CSI_F_DP[0]/ CSI_E_DP[2]	CSI-2 Port F Data lane **OR**	I	DPHY	Diff	Supports X1/X2

continued...

Signal Name	Design Pin Name	Description	Dir.	Buffer Type	Link Type	H/U-Series Processor
	CSI_F_DN[1]/ CSI_E_DN[3] CSI_F_DP[1]/ CSI_E_DP[3]	Continuation of CSI-2 Port E Data lane				*Note:* If CSI_F[1:0] are used as a separate camera CSI_E[3:2] are not supported.
CSI_A_CLK_P CSI_A_CLK_N	CSI_A_CLK_N CSI_A_CLK_P	CSI 2 Port A Clock lane	I	DPHY	Diff	Supported
CSI_B_CLK_P CSI_B_CLK_N	CSI_B_CKN CSI_B_CKP	CSI 2 Port B Clock lane	I	DPHY	Diff	Supported
CSI_E_CLK_P CSI_E_CLK_N	CSI_E_CLK_P CSI_E_CLK_N	CSI 2 Port E Clock lane	I	DPHY	Diff	Supported
CSI_F_CLK_P CSI_F_CLK_N	CSI_F_CLK_P CSI_F_CLK_N	CSI 2 Port F Clock lane	I	DPHY	Diff	Supported
CSI_RCOMP	CSI_RCOMP	CSI Resistance Compensation	Analog	N/A	SE	Supported *Note:* Connected to PHY E

10.0 Intel® Neural Processing Unit (Intel® NPU)

The NPU IP in the Intel® Core™ Ultra Processor configuration is a Deep Learning accelerator enumerated to a host processor as an integrated PCIe device. It delivers the cutting-edge processing throughput required to satisfy the demands of Deep Learning applications. The NPU technology is applicable to personal computing devices such as tablets and PCs as a way to encourage AI based applications and services on power and performance sensitive platforms.

The functionality of the Intel® NPU is exposed to a Host system (enumerated as a PCIe device) via a base set of registers. These registers provide access to control and data path interfaces and reside in the Host and Processor subsystems of the Intel® NPU. All host communications are consumed by the scheduler of the Intel® NPU, a 32-bit LeonRT micro-controller. The LeonRT manages the command and response queues as well as the runtime management of the IP itself.

The NPU IP Deep Learning capability is provided by two Neural Compute Engine (NCE) Tiles. Both NCE Tiles are managed by the NPU Scheduler. Each Tile includes a configurable number of Multiply Accumulate (MAC) engines, purpose built for Deep Learning workloads, and two Intel® Movidius SHAVE DSP processors for optimal processing of custom deep learning operations.

The Intel® NPU of Intel® Core™ Ultra Processor is configured with 2k MACs per tile totaling 4k MACs across both tiles and 4 MB of associated **near compute** memory.

The NPU plugin supports the following data types as inference precision of internal primitives: **INT8(I8/U8), FP16**.

10.1 Functional Description

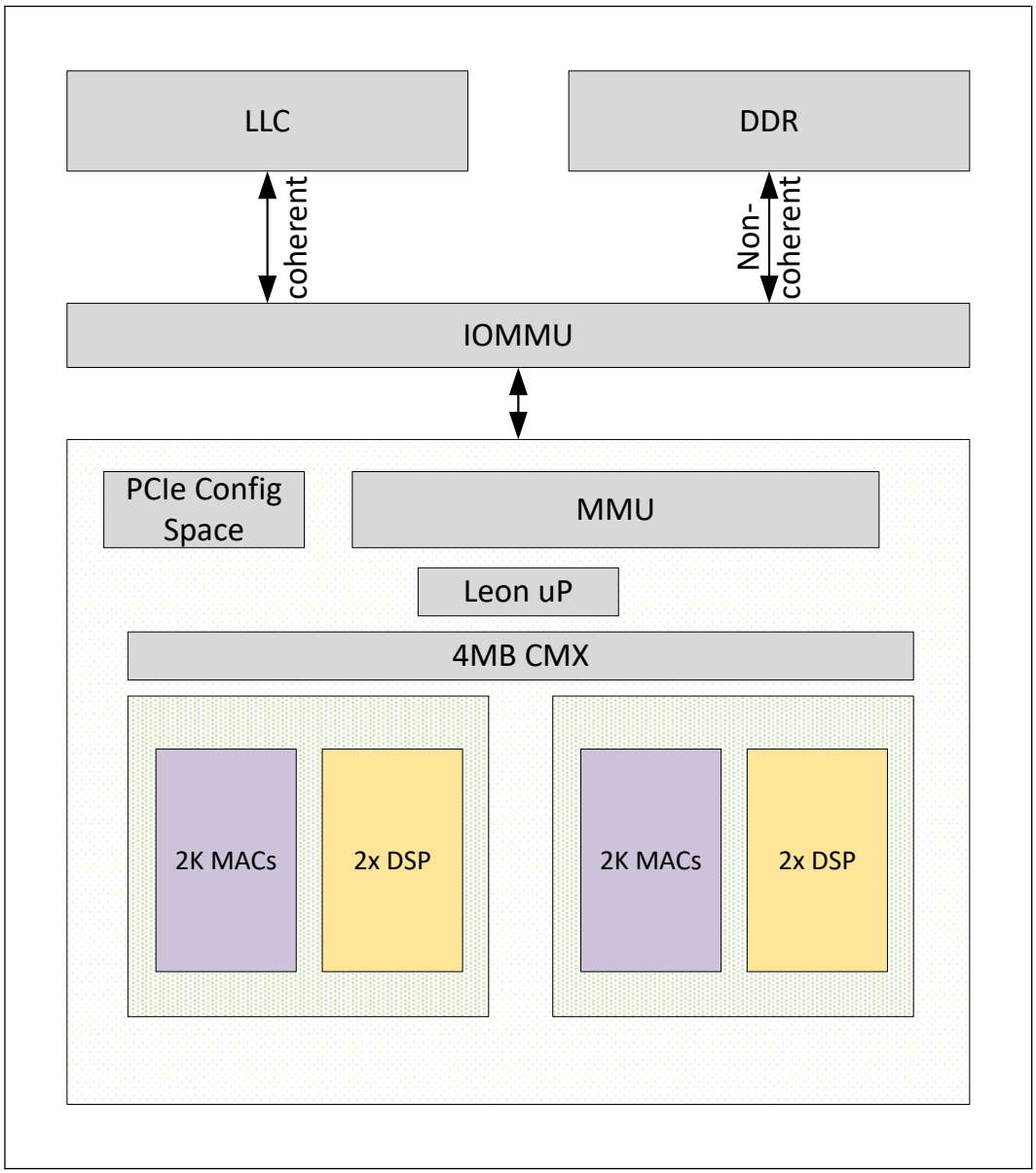
NPU IP comprises 3 subsystems, as follows:

- Processor subsystem
- Host subsystem
- NCE subsystem

Apart from the subsystems, it has a Host interface for data exchange with the system memory. Details of these blocks are provided in the next sections.

Below is the block diagram of NPU IP:

Figure 8. NPU IP Block Diagram



10.1.1 Processor Subsystem

This subsystem provides the SW services through which all functions of the NPU are accessed. Those services are provided by firmware executing on the LeonRT Processor. The LeonRT is the first core out of reset in the NPU (before both the LeonNN and SHAVE cores).

10.1.2 NCE Subsystem

The Neural Compute Subsystem is a hardware accelerator for Deep Neural Network (DNN) workloads. It features a highly configurable pipeline to support DNN (Deep Neural Network) operations such as CNN (Convolutional Neural Networks), LSTM (Long Short-Term memory) and LRN (Local Response Norm). It also leverages activation and weight sparsity optimal performance.

Neural Compute Subsystem is built from up to 2 NCE Tiles (fixed) where each Tile is a primary unit of compute. Each Tile can support 2K Multiply Accumulate circuits (MACs) and two Activation SHAVE Engines (ACTShave). Tiles can be deployed to operate independently across multiple networks (threads) or be aggregated to form a multi cluster engine processing a single network (thread). Refer to the diagram below showing the 4K4M configuration.

NCE Subsystem supports two DMA engines. Each engine supports in-line weight decompression and write data broadcast capability into the local Connection MatriX (CMX) memory (dedicated SRAM).

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

10.1.2.1 Some NCE Subsystem Features

- Dedicated real-time scheduler for job dispatching to DPU and Activation SHAVE engines. This is a LEON core (LeonNN) executing to two levels of cache.
- Two NCE Tiles with 2K MACs per tile.
- Activation SHAVE processors to support custom activation functions. These are vectorized processing units with a 128 bit data bus.
- 2MB of dedicated SRAM memory per tile

10.1.2.2 NCE Tile

The NCE Tile is the building block of the NCE Subsystem. The NCE subsystem supports a fixed two tile configuration. Each NCE tile supports the following:

- Single Data Processing Units (DPU) that supports 2048 MACs built from 512 MAC Processing Engines (MPE) with 4MACs in each MPE.
- An NCE Tile is capable of delivering:
  - 4 TOPS (8-bit) or 2 TFLOPS(FP16) @ 1 GHz[1] DPU Clock Frequency for a single DPU configuration

NOTE

1. 1 GHz is not the maximum frequency of DPU.

- Two ACT-SHAVE DSP with shared data and instruction L2 Cache used for flexible tensor compute operation.

10.1.2.3 ACT-SHAVE

ACT-SHAVE is DSP Processor which supports 128 bit vector operations. Two of ACT-SHAVE DSPs are placed in each NCE Tile and are used for custom layer and standard layers that do not map well to the DPU. All ACT-SHAVE DSP functions shall be included in the graph-file and barriers shall be used for HW Synchronization of the DSP operation and the rest of the schedule.

11.0 Audio Voice and Speech

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

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

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

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

Table 16. Acronyms

Acronyms	Description
DMA	Direct Memory Access.
DMIC	Digital Microphone. PDM based MEMs microphone modules.
DSP	Digital Signal Processor. In AVS specifically a DSP to process audio data.
I^2S	Inter IC Sound. A serial bus using PCM.
MEMs	Micro electrical mechanical Systems. For AVS devices such as Digital MEMs Microphones.
MSI	Message Signaled Interrupt. An in-band method of signaling an interrupt.
PCM	Pulse Code Modulation. Modulation with amplitude coded into stream.
PDM	Pulse Density Modulation. Modulation with amplitude coded by pulse density.
SDI	Serial Data In.
SDO	Serial Data Out.
VOIP	Voice Over Internet Protocol

Table 17. References

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

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

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

- PCI / PCI Express* controller
  - Option to hide PCI configuration space and use ACPI method for enumeration
- Supports data transfers, descriptor fetches, and DMA position writes using VC0 or VC1
- Independent Bus Host logic for 19 general purpose DMA streams: 10 input and 9 output
- Supports variable length stream slots
- Each general purpose stream supports up to:
  - 16 channels per stream
  - 32 bits/sample
  - 192 kHz sample rate
- Supports memory-based command/response transport
- Supports optional Immediate Command/Response mechanism
- Supports input and output stream synchronization
- Supports MSI interrupt delivery
- Support for ACPI D3 and D0 Device States
- Supports Function Level Reset (FLR)
  - Only if exposed as a PCI Express device (or ACPI method)
- Support Converged Platform Power Management (CPPM)
  - Support 1 ms of buffering with all DMA running with maximum bandwidth

11.2 Audio DSP Capabilities

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

- Up to 3 x 393.2 MHz Tensilica* LX7 HIFI4 DSP Cores
- Up to 2816 KB of L2 HP SRAM for each DSP Cores

- L2 uncache memory accessing up to 16 x 16 MB of remote DDR region
- Up to 3 x 8 ch GPDMA for data transfers to and from DSP I/O peripherals
- DSP offload for low power audio rendering and recording
- Various DSP functions provided by Tensillica Core: MP3, AAC, Dolby Digital*, etc.
- Host downloadable DSP FW functions
- Voice call processing enhancement
- HW based DSP accelerators, for example, Machine Learning block and SHA engine

11.3 Intel® High Definition Audio Interface Capabilities

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

- The SDI signals to support external codecs
- Drives variable frequency (6 MHz to 24 MHz) BCLK to support:
  - SDO double pumped up to 48 Mb/s
  - SDIs single pumped up to 24 Mb/s
- Provides cadence for 44.1 kHz-based sample rate output
- Supports LV Mode (1.5 V and 1.8 V)

11.4 Direct Attached Digital Microphone (PDM) Interface

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

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

11.5 USB Audio Offload Support

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

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

11.6 I^2S / PCM Interface

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

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

11.7 Intel® Display Audio Interface

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

- 1 SDI signal to support 1 iDisp audio codec
- Drives 96 MHz frequency BCLK support
  - SDO single pumped to 96 Mb/s

  - SDI single pumped up to 96 Mb/s
- Provides cadence for 44.1 kHz-based sample rate output.

11.8 MIPI® SoundWire* Interface

The SoundWire interface is a feature offering connection to the SoundWire devices, which include audio codecs and modem codecs. The SoundWire interface is the latest audio interface targeting (but not limited to) the phone and tablet market and the main advantage is the connection simplicity with a two wires multi-drop topology and PDM streaming capabilities. SoundWire device class initiative for audio is bringing standardization to the audio codec SW stack. These devices are enumerated based on vendor / device ID of the SoundWire device reporting, allowing vendor customization of audio codec SW if desired. SoundWire interface capabilities are listed as follows:

- Up to 4 SoundWire interfaces frame rate synchronized on global periodic events
- SNDW[0] supports 4 data lane per SoundWire interface
- Support SoundWire Device Class Specification for Audio Controls and Memories
- Up to 8 PCM bidirectional streams per SoundWire interface
  - Direction is programmable as either input or output stream
- Up to 8 channels per PCM streams
- Interrupt / PME wake capable on DATA pin assertion in low power state

NOTE

PDM support exists in design but is not enabled for SNDW links.

11.9 Signal Description

Signal Name	Type	Description
Intel® High Definition Audio Signals		
GPP_D17/**HDA_RST#**/I2S2_RXD/DMIC_DATA1/USB-C_GPP_D17	O	**Intel HD Audio Reset**: Host H/W reset to internal and external codecs.
GPP_D11/**HDA_SYNC**/I2S0_SFRM/DMIC_CLK_B1/USB-C_GPP_D11	O	**Intel HD Audio Sync**: 48 kHz fixed rate frame sync to the codecs.
GPP_D10/**HDA_BCLK**/I2S0_SCLK/DMIC_CLK_A1/USB-C_GPP_D10	O	**Intel HD Audio Bit Clock**: Up to 24 MHz serial data clock generated by the Intel® HD Audio controller.
GPP_D12/**HDA_SDO**/I2S0_TXD/USB-C_GPP_D12	O	**Intel HD Audio Serial Data Out**: Serial TDM data output to the codecs. The serial output is double-pumped for a bit rate of up to 48 Mb/s.
GPP_D13/**HDA_SDI0**/I2S0_RXD/USB-C_GPP_D13	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.

continued...

Signal Name	Type	Description
GPP_D16/**HDA_SDI1**/I2S2_TXD/ DMIC_CLK_B0/USB-C_GPP_D16	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.
I^2S / PCM Interface		
GPP_D10/HDA_BCLK/ **I2S0_SCLK**/DMIC_CLK_A1/USB-C_GPP_D10	I/O	**I^2S / PCM serial bit clock 0**: Serial bit clock used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_S00/SNDW0_CLK/ **I2S1_SCLK**	I/O	**I^2S / PCM serial bit clock 1**:Serial bit clock is used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D14/**I2S2_SCLK**/ DMIC_CLK_A0/USB-C_GPP_D14	I/O	**I^2S / PCM serial bit clock 2**: Serial bit clock is used to control the timing of a transfer. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D11/HDA_SYNC/ **I2S0_SFRM**/DMIC_CLK_B1/USB-C_GPP_D11	I/O	**I^2S / PCM serial frame indicator 0**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_S01/SNDW0_DATA0/ **I2S1_SFRM**	I/O	**I^2S / PCM serial frame indicator 1**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D15/**I2S2_SFRM**/ DMIC_DATA0/USB-C_GPP_D15	I/O	**I^2S / PCM serial frame indicator 1**: This signal indicates the beginning and the end of a serialized data word. Can be generated internally (Host mode) or taken from an external source (Device mode).
GPP_D12/HDA_SDO/**I2S0_TXD**/ USB-C_GPP_D12	O	**I^2S / PCM transmit data (serial data out)0**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_S02/SNDW1_CLK/ SNDW0_DATA1/DMIC_CLK_A0/ **I2S1_TXD**	O	**I^2S / PCM transmit data (serial data out)1**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_D16/HDA_SDI1/**I2S2_TXD**/ DMIC_CLK_B0/USB-C_GPP_D16	O	**I^2S / PCM transmit data (serial data out)1**: Serial data out line. Sample length is a function of the selected serial data sample size.
GPP_D13/HDA_SDI0/**I2S0_RXD**/ USB-C_GPP_D13	I	**I^2S / PCM receive data (serial data in)0**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_S03/SNDW1_DATA/ SNDW0_DATA2/DMIC_DATA0/ **I2S1_RXD**	I	**I^2S / PCM receive data (serial data in)1**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_D17/HDA_RST#/**I2S2_RXD**/ DMIC_DATA1/USB-C_GPP_D17	I	**I^2S / PCM receive data (serial data in)1**: Serial data in line. Sample length is a function of the selected serial data sample size.
GPP_D09/**I2S_MCLK1_OUT**/ USB-C_GPP_D09	O	**I^2S / PCM Host reference clock 0**: This signal is the host reference clock that connects to an audio codec.
DMIC Interface		

continued...

Signal Name	Type	Description
GPP_S02/SNDW1_CLK/ SNDW0_DATA1/**DMIC_CLK_A0**/ I2S1_TXD or GPP_D14/I2S2_SCLK/ **DMIC_CLK_A0**/USB-C_GPP_D14	O	**Digital Mic Clock A0**:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz. Duplication for clock pin (instance A) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance A) should be used.
GPP_S06/SNDW3_CLK/ **DMIC_CLK_A1** or GPP_D10/HDA_BCLK/I2S0_SCLK/ **DMIC_CLK_A1**/USB-C_GPP_D10	O	**Digital Mic Clock A1**:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz. Duplication for clock pin (instance A) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance A) should be used.
GPP_S04/SNDW2_CLK/ SNDW0_DATA3/**DMIC_CLK_B0** or GPP_D16/HDA_SDI1/I2S2_TXD/ **DMIC_CLK_B0**/USB-C_GPP_D16	O	**Digital Mic Clock B0**:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz. Duplication for clock pin (instance B) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance B) can be disconnected.
GPP_S05/SNDW2_DATA/ DMIC_CLK_B1 or GPP_D11/HDA_SYNC/I2S0_SFRM/ **DMIC_CLK_B1**/USB-C_GPP_D11	O	**Digital Mic Clock B**1:Serial data clock generated by the HD Audio controller. The clock output frequency is up to 4.8 MHz. Duplication for clock pin (instance B) in case platform wanted to separate clock connection for left channel mic vs right channel mic. For the case of sharing single clock connection to both left and right channel mics, clock pin (instance B) can be disconnected.
GPP_S03/SNDW1_DATA/ SNDW0_DATA2/**DMIC_DATA0**/ I2S1_RXD or GPP_D15/I2S2_SFRM/ **DMIC_DATA0**/USB-C_GPP_D15	I	**Digital Mic Data**:Serial data input from the digital mic.
GPP_S07/SNDW3_DATA/ **DMIC_DATA1** or GPP_D17/HDA_RST#/I2S2_RXD/ **DMIC_DATA1**/USB-C_GPP_D17	I	**Digital Mic Data**:Serial data input from the digital mic.
SoundWire Interface		
GPP_S00/**SNDW0_CLK**/ I2S1_SCLK	I/O	**SoundWire Clock**: Serial bit clock used to control the timing of a transfer.
GPP_S01/**SNDW0_DATA0**/ I2S1_SFRM	I/O	**SoundWire Data**: Serialized data line containing framing and data being transmitted/received.
GPP_S02/**SNDW1_CLK**/ **SNDW0_DATA1**/DMIC_CLK_A0/ I2S1_TXD	I/O	**SoundWire Clock**: Serial bit clock used to control the timing of a transfer. **SoundWire Data**: Serialized data line containing framing and data being transmitted/ received.
GPP_S03/**SNDW1_DATA**/ **SNDW0_DATA2**/DMIC_DATA0/ I2S1_RXD	I/O	**SoundWire Data**: Serialized data line containing framing and data being transmitted/received.

continued...

Signal Name	Type	Description
GPP_S04/**SNDW2_CLK**/ **SNDW0_DATA3**/DMIC_CLK_B0	I/O	**SoundWire Clock**: Serial bit clock used to control the timing of a transfer. **SoundWire Data**: Serialized data line containing framing and data being transmitted / received.
GPP_S05/**SNDW2_DATA**/ DMIC_CLK_B1	I/O	**SoundWire Data**: Serialized data line containing framing and data being transmitted / received.
GPP_S06/**SNDW3_CLK**/ DMIC_CLK_A1	I/O	**SoundWire Clock**: Serial bit clock used to control the timing of a transfer.
GPP_S07/**SNDW3_DATA**/ DMIC_DATA1	I/O	**SoundWire Data**: Serialized data line containing framing and data being transmitted / received.
SNDW_RCOMP	A	SoundWire Resistor compensation.

11.10 Integrated Pull-Ups and Pull-Downs

Table 18. Integrated Pull-Ups and Pull-Downs

Signal Name	Resistor Type	Value
HDA_SYNC	Pull-down	20 kohm
HDA_SDO	Pull-down	20 kohm
HDA_SDI[1:0]	Pull-down	20 kohm
I2S[2:0]_SCLK	Pull-down	20 kohm
I2S[2:0]_SFRM	Pull-down	20 kohm
I2S[2:0]_RXD	Pull-down	20 kohm
DMIC_DATA[1:0]	Pull-down	20 kohm
SNDW0_DATA[3:0]	Pull-down	5 kohm
SNDW[3:1]_DATA	Pull-down	5 kohm

11.11 I/O Signal Planes and States

Table 19. I/O Signal Planes and States

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S4/S5
High Definition Audio Interface				
HDA_RST#	Primary	Asserted	Asserted	Asserted
HDA_SYNC	Primary	Internal Pull-down	Driven Low	Internal Pull-down
HDA_BCLK	Primary	Driven Low	Driven Low	Driven Low
HDA_SDO	Primary	Internal Pull-down	Driven Low	Internal Pull-down
HDA_SDI[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
I²S/PCM Interface				
I2S[2:0]_SCLK	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
I2S[2:0]_SFRM	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down

continued...

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S4/S5
I2S[2:0]_TXD	Primary	Driven Low	Driven Low	Driven Low
I2S[2:0]_RXD	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
I2S_MCLK1_OUT	Primary	Driven Low	Driven Low	Driven Low
DMIC Interface				
DMIC_CLK_A[1:0]	Primary	Driven Low	Driven Low	Driven Low
DMIC_CLK_B[1:0]	Primary	Driven Low	Driven Low	Driven Low
DMIC_DATA[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SoundWire Interface				
SNDW0_DATA[3:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SNDW[3:1]_DATA	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SNDW[3:0]_CLK	Primary	Driven Low	Driven Low	Driven Low
Note: 1. Pull-down is enabled with PLTRST# is asserted for the following signals: HDA_SYNC, HDA_SDO, HDAPROC_SDO				

12.0 Power Management

Table 20. Acronyms

Acronyms	Description
VR	Voltage Regulator

Table 21. References

Specification	Location
Advanced Configuration and Power Interface (ACPI)	https://uefi.org/htmlspecs/ACPI_Spec_6_4_html/04_ACPI_Hardware_Specification/ACPI_Hardware_Specification.html

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

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

Table 22. General System Power States

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

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

NOTE

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

Table 23. State Transition Rules for the Processor

Present State	Transition Trigger	Next State
G0/S0/C0	• SLP_EN bit set • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/Cx • G1/S4, or G2/S5 state • G2/S5 • G3
G0/S0/Cx	• Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0 • S5 • G3
G1/S4	• Any Enabled Wake Event • Power Button Override[3] • Mechanical Off/Power Failure	• G0/S0/C0[2] • G2/S5 • G3
G2/S5	• Any Enabled Wake Event • Mechanical Off/Power Failure	• G0/S0/C0[2] • G3
G2	• Any Enabled Wake Event • Mechanical Off/Power Failure • Power Button Override	• G0/S0/C0[2] • G1/S4 or G2/S5 • G3 • G2/S5
G3	• Power Returns	• S0/C0 (reboot) or G2/S5[4] (stay off until power button pressed or other wake event)[1,2]

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

System Power Planes

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

NOTE

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

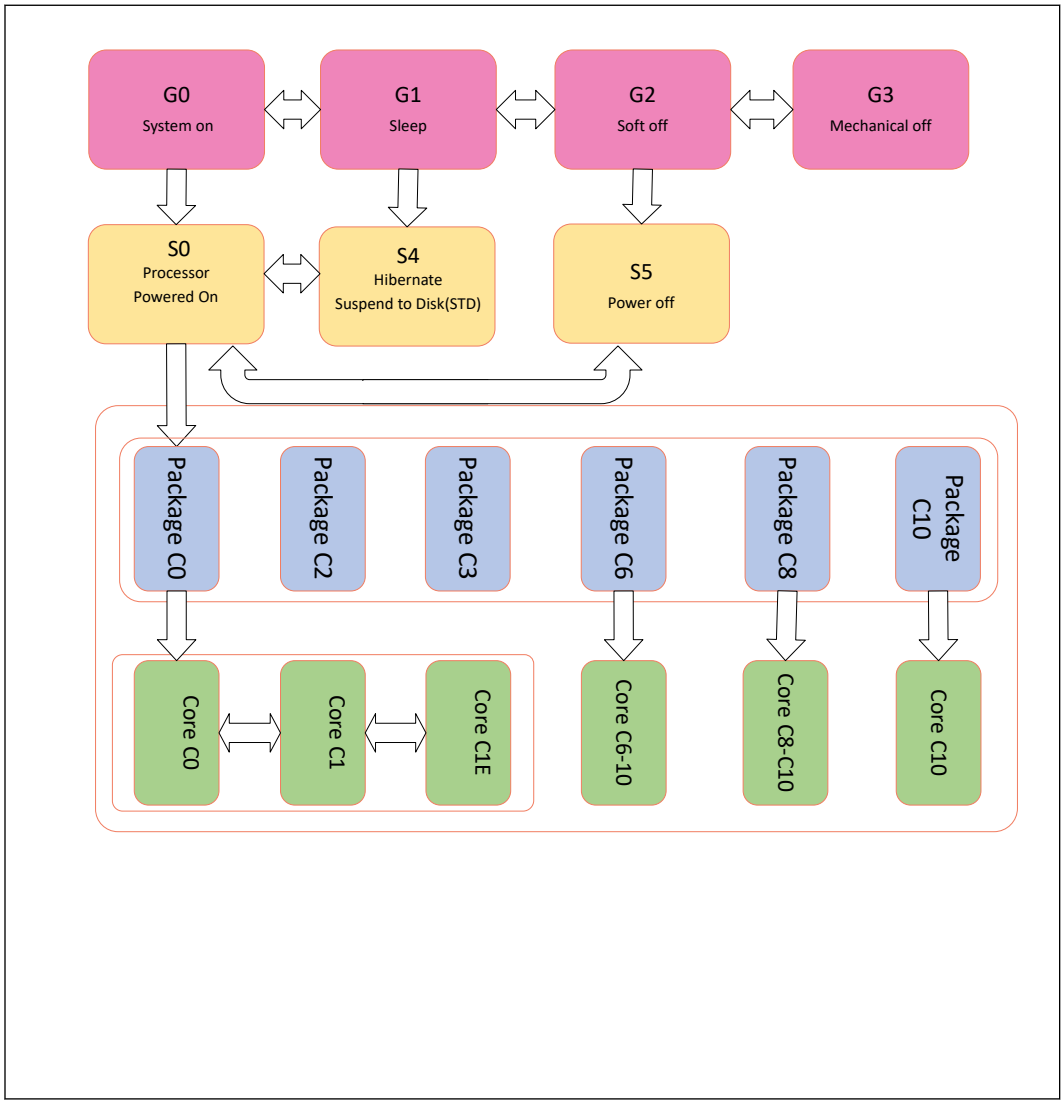
Table 24. System Power Plane

Plane	Controlled By	Description
Memory	SLP_S4# signal SLP_S5# signal	When SLP_S4# goes active, power can be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory can also be shut down.

continued...

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

Figure 9. Power State Block Diagram



12.2 Legacy Power Management Support

12.2.1 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 processor implements an ALT access mode.

Write Only Registers with Read Paths in ALT Access Mode

The registers described in below table 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 25. Write Only Registers with Read Paths in ALT Access Mode

Restore Data			
I/O Addr	**# of Rds**	**Access**	**Data**
20h	12	1	PIC ICW2 of Primary controller
		2	PIC ICW3 of Primary controller
		3	PIC ICW4 of Primary controller
		4	PIC OCW1 of Primary controller [1]
		5	PIC OCW2 of Primary controller
		6	PIC OCW3 of Primary controller
		7	PIC ICW2 of Secondary controller
		8	PIC ICW3 of Secondary controller
		9	PIC ICW4 of Secondary controller
		10	PIC OCW1 of Secondary controller [1]
		11	PIC OCW2 of Secondary controller
		12	PIC OCW3 of Secondary controller
40h	7	1	Timer Counter 0 status, bits [5:0]
		2	Timer Counter 0 base count low byte
		3	Timer Counter 0 base count high byte
		6	Timer Counter 2 base count low byte
		7	Timer Counter 2 base count high byte
42h	1		Timer Counter 2 status, bits [5:0]
70h	1		Bit 7 = Read value is ‘0’. Bits [6:0] = RTC Address

Notes: 1. The OCW1 register must be read before entering ALT access mode.
2. Bits 5, 3, 1, and 0 return 0.

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 below.

Table 26. 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

12.2.2 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 processor does not support burst modes.

Mobile APM Power Management

In mobile systems, there are additional requirements associated with device power management. To handle this, the processor 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.

12.3 Functional Description

12.3.1 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 S4 state – Suspend-to-Disk (STD)
  - ACPI G2/S5 state – Soft Off (SOFF)
  - Power Failure Detection and Recovery
- Intel® CSME Power Management Support
  - Wake events from the Intel® CSME (enabled from all S-States including Catastrophic S5 conditions)

12.3.2 Power Saving Features

Power Management Substates

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

S0ix in Sx

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

Naming Convention

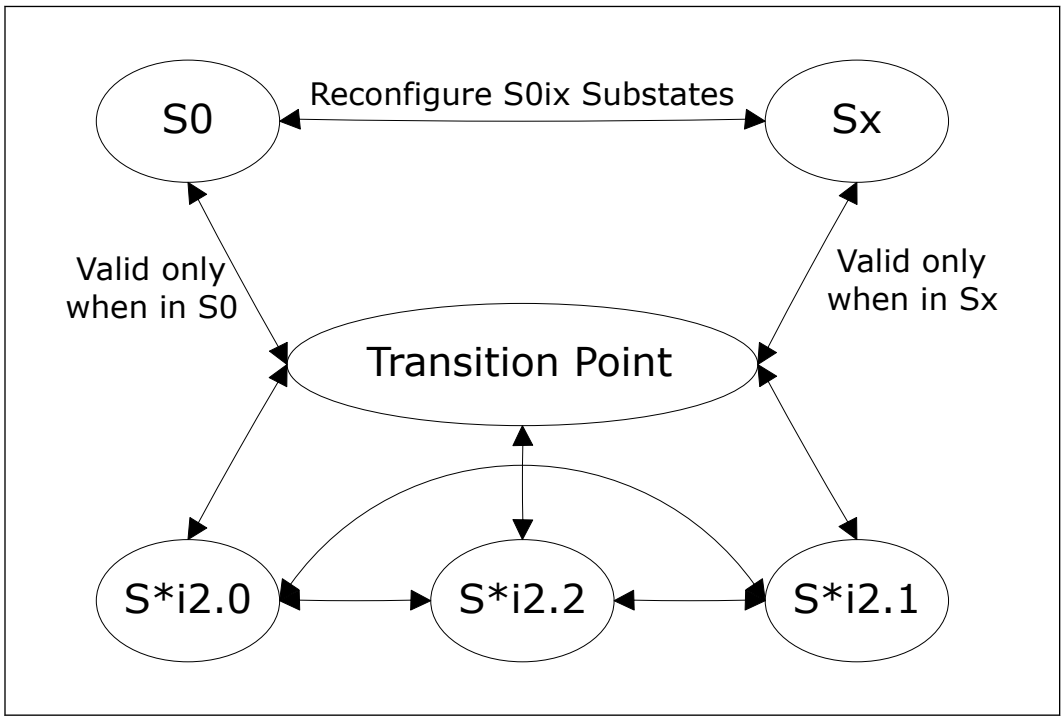
The naming convention: S*ix.y refers to any combination of S0/Sx and Substate.

Specifically:

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

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

Figure 10. Power Management Substates



12.3.3 SMI#/SCI Generation

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

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

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

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

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

NOTE

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

Table 27. Causes of SMI and SCI

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

continued...

Cause	SCI	SMI	Additional Enables[1]	Where Reported
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 [9]	Yes	Yes	eSPI_SCI_EN	eSPI_SCI_STS eSPI_SMI_STS
Software Generated GPE	Yes	Yes	SWGPE_EN=1	SWGPE_STS
Intel® CSME	Yes	Yes	CSME_SCI_EN=1 CSME_SCI_EN=0; CSME_SMI_EN=1;	CSME_SCI_STS CSME_SMI_STS
GPIO Lockdown Enable bit changes from '1' to '0'	No	Yes	GPIO_UNLOCK_SMI_EN=1	GPIO_UNLOCK_SMI_STS
USB 3.2 (xHCI) SMI Event	No	Yes	xHCI_SMI_EN=1	xHCI_SMI_STS
Wake Alarm Device Timer	Yes	Yes	WADT_EN	WADT_STS
ISH	Yes	No	ISH_EN	ISH_STS
RTC update-in-progress	No	Yes	Refer to Vol2	RTC_UIP_SMI_STS
SIO SMI events	No	Yes	SIP_SMI_EN	SIO_SMI_STS
SCC	No	Yes	SCC_SMI_EN	SCC_SMI_STS

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

PCI Express* SCI

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

PCI Express* Hot-Plug

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

12.3.4 Sleep States

Sleep State Overview

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

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

Initiating Sleep State

Sleep states (S4/S5) are initiated by:

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

Table 28. Sleep Types

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

Exiting Sleep States

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

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

NOTE

If the BATLOW# signal is asserted, the processor does not attempt to wake from an S4/S5 state, 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 processor, and the system wakes after BATLOW# is de-asserted.

Table 29. Causes of Wake Events

Cause	How Enabled	Wake from Sx	Wake from Sx After Power Loss[2]	Wake from "Reset" Types[3]
RTC Alarm	Set RTC_EN bit in PM1_EN_STS register.	Yes	Yes	No
Power Button	Always enabled as Wake event.	Yes	Yes	Yes
Any GPIOs except DSW GPIOs can be enabled for wake	Refer to Note 5	Yes	No	No
LAN_WAKE#	Enabled natively (unless pin is configured to be in GPIO mode)	Yes	Yes	Yes
Intel® High Definition Audio	Event sets PME_B0_STS bit; PM_B0_EN must be enabled. Cannot wake from S5 state if it was entered due to power failure or power button override.	Yes	Yes	No
Primary PME#	PME_B0_EN bit in GPE0_EN[127:96] register.	Yes	Yes	No
Secondary PME#	Set PME_EN bit in GPE0_EN[127:96] register.	Yes	Yes	No
PCI Express* WAKE# pin	PCIEXP_WAKE_DIS bit.	Yes	Yes	No
SMBALERT#	Refer to Note 4	Yes	Yes	Yes
SMBus Target Wake Message (01h)	Wake/SMI# command always enabled as a Wake event. *Note:* SMBus Target Message can wake the system from S4/S5, as well as from S5 due to Power Button Override.	Yes	Yes	Yes
SMBus Host Notify message received	HOST_NOTIFY_WKEN bit SMBus Target Command register. Reported in the SMB_WAK_STS bit in the GPE0_STS register.	Yes	Yes	Yes
Intel® CSME Non-Maskable Wake	Always enabled as a wake event.	Yes	Yes	Yes
Integrated WoL Enable Override	WoL Enable Override bit (in Configuration Space).	Yes	Yes	Yes
Wake Alarm Device	WADT_EN in GPE0_EN[127:96]	Yes	No	No

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

PCI Express* WAKE# Signal and PME Event Message

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

NOTE

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

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

Sx-G3-Sx, Handling Power Failures

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

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

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

Table 30. Transitions Due to Power Failure

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

12.3.5 Event Input Signals and Their Usage

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

PWRBTN# (Power Button)

The PWRBTN# signal operates as a "Fixed Power Button" as described in the *Advanced Configuration and Power Interface Specification.* PWRBTN# signal has a 16 ms de-bounce on the input. The state transition descriptions are included in the below table.

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

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

The processor also supports modifying the length of time the Power Button must remain asserted before the unconditional power down occurs (4-14 seconds). The length of the Power Button override duration has no impact on the "extension" of the power button override timer while SLP_* stretching is in progress. The extended power button override period while stretching is in progress remains 9-10 seconds in all cases.

Table 31. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI or SCI generated (depending on SCI_EN, PWRBTN_EN and GLB_SMI_EN)	Software typically initiates a Sleep state *Note:* Processing of transitions starts within 100 us of the PWRBTN# input pin to processor going low.[1]
S5	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup *Note:* Could be impacted by SLP_* min assertion. The minimum time the PWRBTN# pin should be asserted is 150 us. The processor will start processing this change once the minimum time requirement is satisfied.[1]
G3	PWRBTN# pressed	None	No effect since no power Not latched nor detected *Notes:* 1. During G3 exit, PWRBTN# pin must be kept de-asserted for a minimum time of 500 us after the RSMRST# has de-asserted.[2] 2. Beyond this point, the minimum time the PWRBTN# pin has to be asserted to be registered by processor as a valid wake event is 150 us.[1]
S0 – S4	PWRBTN# held low for at least 4 3 consecutive seconds	Unconditional transition to S5 state.	No dependence on processor or any other subsystem

continued...

Present State	Event	Transition/Action	Comment
			Note: Due to internal processor latency, it could take up to an additional ~1.3s after PWRBTN# has been held low for 4s before the system would begin transitioning to S5.
Notes: 1. If PM_CFG.PB_DB_MODE='0', the debounce logic adds 16 ms to the start/minimum time for processing of power button assertions. 2. This minimum time is independent of the PM_CFG.PB_DB_MODE value. 3. The amount of time PWRBTN# must be asserted is configurable via PM_CFG2.PBOP. 4 seconds is the default.			

Power Button Override Function

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

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

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

NOTE

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

Sleep Button

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

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

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_B0_STS bit that will be set by the processor 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.

SYS_RESET# Signal

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

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

NOTES

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

THERMTRIP# Signal

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

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

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

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

processor must only honor the THERMTRIP# pin while it is being driven to a valid state by the processor. The THERMTRIP# Valid Point ='0', implies processor will start monitoring THERMTRIP# at PLTRST# de-assertion (default). The THERMTRIP# Valid

Point ='1', implies processor will start monitoring THERMTRIP# at PLT_PWROK assertion. Regardless of the setting, the processor must stop monitoring THERMTRIP# at PLT_PWROK de-assertion.

NOTE

A thermal trip event will clear the PWRBTN_STS bit.

Sx_Exit_Holdoff#

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

Requirements to support Sx_Exit_Holdoff#

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

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

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

NOTE

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

12.3.6 System Power Supplies, Planes, and Signals

Power Plane Control

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

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® Converged Security and Management Engine and SPI flash on a platform that supports the M3 state (for example, certain power policies in Intel® AMT).

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

SLP_S4# and 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 processor provides a mechanism to fully cycle the power to the DRAM and/or detect if the power is not cycled for a minimum time.

NOTE

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

PLT_PWROK Signal

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

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

NOTE

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

BATLOW# (Battery Low)

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

SLP_LAN# Pin Behavior

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

- The LAN PHY is always powered when the Host and Intel® CSME systems are running.
  - SLP_LAN#='1' whenever SLP_S3#='1' or SLP_A#='1'.
- If the LAN PHY is required by Intel® CSME in Sx/M-Off, Intel® CSME must configure SLP_LAN#='1' irrespective of the power source and the destination power state. Intel® CSME must be powered at least once after G3 to configure this.
- If the LAN PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN.
- If the LAN PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN.
- If the LAN PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS.

NOTE

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

SLP_WLAN# Pin Behavior

The processor 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 S4/S5 states, the host BIOS must set HOST_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 3.3 V power rail.

PRIMPWRDNACK Steady State Pin Behavior

Below table summarizesPRIMPWRDNACK pin behavior.

Table 32. PRIMPWRDNACK/GPP_A02 Pin Behavior

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

Table 33. PRIMPWRDNACK During Reset

Reset Type (Note)	SPDA Value
Power-cycle Reset	0
Global Reset	0
Straight to S5	Processor initially drive '0' and then drive per Intel® CSME power policy configuration.
Note: Refer to Table 34 on page 102	

RTCRST# and SRTCRST#

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

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

PROC_C10_GATE#

When asserted, PROC_C10_GATE# is the indication to the system that the processor is entering C10.

12.3.7 Reset Behavior

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

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

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

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

Straight to S5 is another reset type where all power wells that are controlled by the SLP_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 processor functionality is reset except RTC Power Well backed information and Suspend well status, configuration, and functional logic for controlling and reporting the reset. The host stays there until a valid wake event occurs.

The following table shows the various reset triggers.

Table 34. Causes of Host and Global Resets

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

continued...

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
Power Button 4 second override causes transition to S5 and reset asserts	No	No	No	Yes
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 1	No	No	Yes	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No[4]	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0 and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No[4]	
Intel® Converged Security and Management Engine Triggered Host Reset without Power-Cycle	Yes	No	No[4]	
Intel® Converged Security and Management Engine Triggered Host Reset with Power-Cycle	No	Yes	No[4]	
Intel® Converged Security and Management Engine Triggered Power Button Override	No	No	No	Yes
Intel® Converged Security and Management Engine Watchdog Timer Timeout	No	No	No[7]	Yes
Intel® Converged Security and Management Engine Triggered Global Reset	No	No	Yes	
Intel® Converged Security and Management Engine Triggered Host Reset with power down (host stays there)	No	Yes[5]	No[4]	
PLTRST# Entry Timeout (Note 6)	No	No	Yes	
PLT_PWROK Stuck Low	No	No	Yes	
Power Management Watchdog Timer	No	No	No[7]	Yes
Intel® Converged Security and Management Engine Hardware Uncorrectable Error	No	No	No[7]	Yes

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

12.4 Processor IA Core Power Management

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

12.4.1 OS/HW Controlled P-states

12.4.1.1 Enhanced Intel SpeedStep® Technology

Enhanced Intel SpeedStep® Technology enables OS to control and select P-state. For more information, refer to Enhanced Intel SpeedStep® Technology on page 119.

12.4.1.2 Intel® Speed Shift Technology

Intel® Speed Shift Technology is an energy efficient method of frequency control by the hardware rather than relying on OS control. For more details, refer to Intel® Speed Shift Technology on page 120.

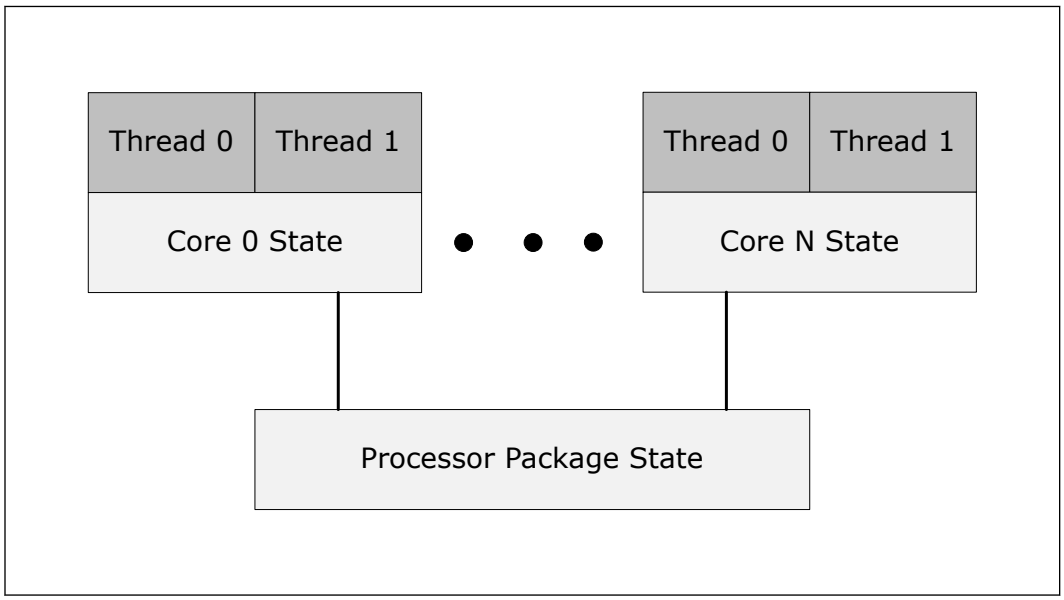
12.4.2 Low Power Idle States

When the processor is idle, low-power idle states (C-states) are used to save power. More power savings actions are taken for numerically higher C-states. However, deeper C-states have longer exit and entry latencies. Resolution of C-states occurs at the thread, processor IA core, and processor package level. Thread-level C-States are available if Hyper-Threading Technology is enabled and the processor IA core support multiple threads.

CAUTION

Long-term reliability cannot be assured unless all the Low-Power Idle States are enabled.

Figure 11. Idle Power Management Breakdown of the Processor IA Cores



While individual threads can request low-power C-states, power saving actions only take place once the processor IA core C-state is resolved. processor IA core C-states are automatically resolved by the processor. For thread and processor IA core C-states, a transition to and from C0 state is required before entering any other C-state.

12.4.3 Requesting the Low-Power Idle States

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

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

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

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

12.4.4 Processor IA Core C-State Rules

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

- A processor IA core C-State is determined by the lowest numerical thread state (such as Thread 0 requests C1E while Thread 1 requests C6 state, resulting in a processor IA core C1E state). Refer to the *G, S, and C Interface State Combinations* table.
- A processor IA core transitions to C0 state when:
  - An interrupt occurs
  - There is an access to the monitored address if the state was entered using an MWAIT/Timed MWAIT instruction
  - The deadline corresponding to the Timed MWAIT instruction expires
- An interrupt directed toward a single thread wakes up only that thread.
- If any thread in a processor IA core is active (in C0 state), the core's C-state will resolve to C0.
- Any interrupt coming into the processor package may wake any processor IA core.
- A system reset re-initializes all processor IA cores.

Table 35. Core C-states

Core C-State	C-State Request Instruction	Description
C0	N/A	The normal operating state of a processor IA core where a code is being executed
C1	MWAIT(C1)	AutoHalt - core execution stopped, autonomous clock gating (package in C0 state)
C1E	MWAIT(C1E)	Core C1 + lowest frequency and voltage operating point (package in C0 state)
C6-C10	MWAIT(C6/C8/10) or IO read=P_LVL3//6/8	Processor IA, flush their L1 instruction cache, the L1 data cache, and L2 cache to the LLC shared cache cores save their architectural state to an SRAM before reducing IA cores voltage, if possible may also be reduced to 0V. Core clocks are off.

This feature is disabled by default. BIOS should enable it in the PMG_CST_CONFIG_CONTROL register. The auto-demotion policy is also configured by this register.

12.4.5 Package C-States

The processor supports C0, C2, C3, C6, C8, and C10 package states. The following is a summary of the general rules for package C-state entry. These apply to all package C-states, unless specified otherwise:

- A package C-state request is determined by the lowest numerical processor IA core C-state amongst all processor IA cores.
- A package C-state is automatically resolved by the processor depending on the processor IA core idle power states and the status of the platform components.
  - Each processor IA core can be at a lower idle power state than the package if the platform does not grant the processor permission to enter a requested package C-state.
  - The platform may allow additional power savings to be realized in the processor.

  - For package C-states, the processor is not required to enter C0 before entering any other C-state.
  - Entry into a package C-state may be subject to auto-demotion – that is, the processor may keep the package in a deeper package C-state then requested by the operating system if the processor determines, using heuristics, that the deeper C-state results in better power/performance.

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

- If a processor IA core break event is received, the target processor IA core is activated and the break event message is forwarded to the target processor IA core.
  - If the break event is not masked, the target processor IA core enters the processor IA core C0 state and the processor enters package C0.
  - If the break event is masked, the processor attempts to re-enter its previous package state.
- If the break event was due to a memory access or snoop request,
  - But the platform did not request to keep the processor in a higher package C-state, the package returns to its previous C-state.
  - And the platform requests a higher power C-state, the memory access or snoop request is serviced and the package remains in the higher power C-state.

Table 36. Package C-States

Package C state	Description	Dependencies
PKG C0	Processor active state. At least one IA core in C0. Processor Graphic in RC0 (Graphics active state) or RC6 (Graphics Core power down state).	-
PKG C2	Cannot be requested explicitly by the Software. All processor IA cores in C6 or deeper + Processor Graphic cores in RC6, memory path may be open. The processor will enter Package C2 when: • Transitioning from Package C0 to deep Package C state or from deep Package C state to Package C0. • All IA cores requested C6 or deeper + Processor Graphic cores in RC6 but there are constraints (LTR, programmed timer events in the near future and so forth) prevent entry to any state deeper than C2 state. • All IA cores requested C6 or deeper + Processor Graphic cores in RC6 but a device memory access request is received. Upon completion of all outstanding memory requests, the processor transitions back into a deeper package C-state.	All processor IA cores in C6 or deeper. Processor Graphic cores in RC6.
PKG C3	Cannot be requested explicitly by the Software. All cores in C6 or deeper + Processor Graphics in RC6, LLC may be flushed and turned off, memory in self refresh, memory clock stopped. The processor will enter Package C3 when: • All IA cores in C6 or deeper + Processor Graphic cores in RC6. • The platform components/devices allows proper LTR for entering Package C3.	All processor IA cores in C6 or deeper. Processor Graphics in RC6. memory in self refresh, memory clock stopped. LLC may be flushed and turned off.
PKG C6	Package C3 + BCLK is off + IMVP VRs voltage reduction/PSx state is possible. The processor will enter Package C6 when:	Package C3. BCLK is off.

continued...

Package C state	Description	Dependencies
	• All IA cores in C6 or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C6.	IMVP VRs voltage reduction/PSx state is possible.
PKG C8	Of all IA cores requested C8 + LLC should be flushed at once, voltage will be removed from the LLC. The processor will enter Package C8 when: • All IA cores in C8 or deeper + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C8.	Package C6 If all IA cores requested C8, LLC is flushed in a single step, voltage will be removed from the LLC.
PKG C10	Package C8 + display in PSR or powered, ff all VRs at PS4 or LPM + crystal clock off. The processor will enter Package C10 when: • All IA cores in C10 + Processor Graphic cores in RC6. • The platform components/devices allow proper LTR for entering Package C10.	Package C8. All IA cores in C8 or deeper. Display in PSR or powered off[1]. All VRs at PS4 or LPM. Crystal clock off.
Note: Display In PSR is only on single embedded panel configuration and panel support PSR feature.		

Package C-State Auto-Demotion

The Processor may demote the Package C-State to a shallower Package C-State to enable better performance, for example instead of going into Package C10, processor will demote to Package C6 (and shallower as required).

The processor's decision to demote the Package C-State is based on Power management parameters such as required C states latencies, entry/exit energy/power, Core wake rates, and device LTR (Latency Tolerance Report). This means that the processor is optimized to minimize platform energy for scenarios with low idle time.

Processor deeper Package C-State entry frequency is controlled to minimize platform energy. When Package C-State Auto-Demotion enabled, a reduced residency in a deeper Package C-State is expected during system runs with high wake rates. For example, some USB/Bluetooth* audio devices may request high wake rates to keep audio quality of service, this audio behavior may result in Package C-State Demotion and impact power consumption.

No change at IDLE scenario power consumption due to this feature. Package C-State Auto-Demotion is enabled by default and controlled through BIOS menu.

Modern Standby

Modern Standby is a platform state. On display time out the OS requests the processor to enter package C10 and platform devices at RTD3 (or disabled) in order to attain low power in idle. Modern Standby requires proper BIOS and OS configuration.

Dynamic LLC Sizing

When all processor IA cores request C8 or deeper C-state, internal heuristics dynamically flushes the LLC. Once the processor IA cores enter a deep C-state, depending on their MWAIT sub-state request, the LLC is either gradually flushed N-ways at a time or flushed all at once. Upon the processor IA cores exiting to C0 state, the LLC is gradually expanded based on internal heuristics.

12.4.6 Package C-States and Display Resolutions

The integrated graphics engine has the frame buffer located in system memory. When the display is updated, the graphics engine fetches display data from system memory. Different screen resolutions and refresh rates have different memory latency requirements. These requirements may limit the deepest Package C-state the processor can enter. Other elements that may affect the deepest Package C-state available are the following:

- Display is on or off
- Single or multiple displays
- Native or non-native resolution
- Panel Self Refresh (PSR) technology

NOTE

Display resolution is not the only factor influencing the deepest Package C-state the processor can get into. Device latencies, interrupt response latencies, and core C-states are among other factors that influence the final package C-state the processor can enter.

The following table lists display resolutions and deepest available package C-State. The display resolutions are examples using common values for blanking and pixel rate. Actual results will vary. The table shows the deepest possible Package C-state. System workload, system idle, and AC or DC power also affect the deepest possible Package C-state.

Table 37. Deepest Package C-State Available

		U/H Processor Series	
Resolution	**Number of Displays**	**PSR Enabled**	**PSR Disabled**
Up to 5120x3200 60Hz[3]	Single	PC10	PC8

Notes: 1. All Deep states are with Display ON.
2. The deepest C-state has variance, dependent various parameters such SW and Platform devices.
3. Partial data based on Pre-Silicon estimation, expected to be update in a future Datasheet release.

12.5 Processor Graphics Power Management

12.5.1 Memory Power Savings Technologies

Intel® Rapid Memory Power Management (Intel® RMPM)

Intel® Rapid Memory Power Management (Intel® RMPM) conditionally places memory into self-refresh when the processor is in package C3 or deeper power state to allow the system to remain in the deeper power states longer for memory not reserved for graphics memory. Intel® RMPM functionality depends on graphics/display state (relevant only when processor graphics is being used), as well as memory traffic patterns generated by other connected I/O devices.

12.5.2 Display Power Savings Technologies

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

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

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

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

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

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

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

Intel® Low Refresh Rate (Intel® LRR)

Intel® LRR is combination of PSR2 and Dynamic Refresh Rate Switching. Intel® LRR uses two mechanisms for switching the refresh rate:

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

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

Panel Self-Refresh 2 (PSR 2)

Panel Self-Refresh feature allows the Processor Graphics core to enter low-power state when the frame buffer content is not changing constantly. This feature is available on panels capable of supporting Panel Self-Refresh. PSR 2 adds partial frame updates and requires an compliant panel.

Low-Power Single Pipe (LPSP)

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

Low-Power Dual Pipe (LPDP)

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

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

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

Intel® S2DDT is most effective with:

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

12.5.3 Processor Graphics Core Power Savings Technologies

Intel® Graphics Dynamic Frequency

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

Intel® Graphics Render Standby Technology (Intel® GRST)

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

Intel Capped Frames Per Second (CFPS)

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

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

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

12.6 TCSS Power State

Table 38. TCSS Power State

TCSS Power State	Processor PM State	Device Attached	Description
TC0	S0	Yes	xHCI, xDCI, USB4 controllers may be active. USB4 DMA / PCIe may be active.
TC7	S*i2.1	Yes	xHCI and xDCI are in D3. USB4 controller is in D3 or D0 idle. USB4 PCIe is inactive.
TC10	S*i2.2	No	Deepest Power state. xHCI / xDCI / USB4 controller are in D3. USB4 DMA / USB4 PCIe are in D3. IOM is in low power state.
"S*i2.1/S*i2.1" - See Naming Convention in Power Saving Features chapter. IOM - TCSS Input Output Manager: • The IOM interacts with the processor to perform power management, boot, reset, connect and disconnect devices to TYPE-C sub-system. TCSS Devices (xHCI / xDCI / TBT Controllers) - power states: • D0 - Device at Active state. • D3 - Device at lowest-powered state.			

12.7 Power and Performance Technologies

12.7.1 Intel® Thread Director

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

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

- Monitors the runtime instruction mix of each thread and the state of each core with nanosecond precision.
- Provides runtime feedback to the OS to make the optimal decision for any workload.

- Dynamically adapts its guidance according to the Assured Based Power (ABP) of the system, operating conditions, and power settings.

12.7.2 Intel® Smart Cache Technology

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

- The LLC is shared between all Compute tile cores (of any type). The maximal size of LLC is 3MB (12 ways, set associative) per P-core or E-core module (bundle of 4 E-Cores).
- The LLC is non-inclusive.
- The LLC may also be referred to as a 3rd level cache.

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

U-Processor Series

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

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

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

On the data side, it is built as one-level cache, with L1 of 48KB, 12-way set-associative.

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

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

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

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

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

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

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

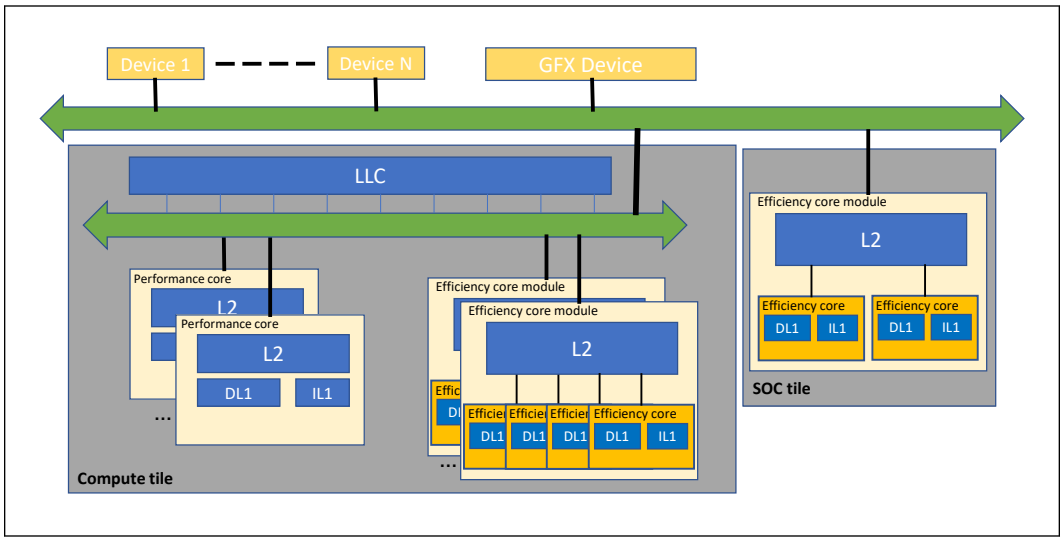
On the data side, it is built as one-level cache, with L1 of 32KB, 8-way set-associative.

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

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

- The P-core 2nd level caches are not shared between physical cores and each physical core has a separate set of caches. Its size is 2MB and it is a 16-way associative non-inclusive cache.
- The E-core 2nd level cache is shared between E-Cores within a module of 4 E-Cores in the Compute tile. Its size is 2MB and it is a 16-way associative non-inclusive cache.
- The LP E-core 2nd level cache is shared between LP E-Cores within a module of 2 LP E-Cores in the SOC tile. Its size is 2MB and it is a 16-way associative non-inclusive cache.

Figure 12. U-Processor P-core, E-core, and LP E-core Cache Hierarchy



NOTE

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

Cache	P-core	E-Core	LP E-core
L1 D L1	48KB 12-way set-associative per core	32KB 8-way set-associative per core	32KB 8-way set-associative per core
L1 I L1	64KB 16-way set-associative per core	64KB 16-way set-associative per core	64KB 16-way set-associative per core
L2	2MB 16-way set-associative per core	2MB 16-way set-associative within a module of 4 Compute tile E-cores	2MB 16-way set-associative within a module of 2 SOC tile LP E-cores
L3	Maximum of 3 MB per P-core / module of 4 E-cores shared across Compute tile		None

H-Processor Series

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

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

- A Data Cache (DL0, DL1)

- An Instruction Cache (IL1)

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

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

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

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

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

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

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

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

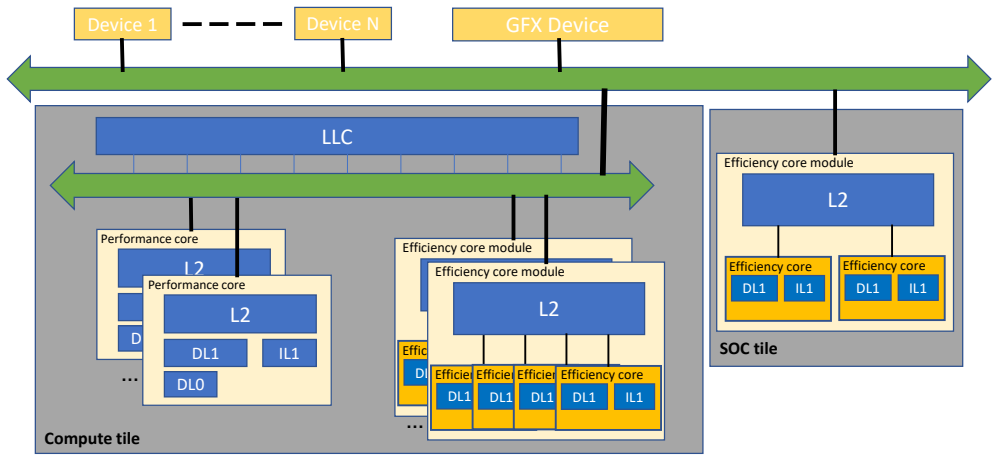
On the data side, it is built as one-level cache, with L1 of 32KB, 8-way set-associative.

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

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

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

Figure 13. H-Processor P-core, E-core, and LP E-core Cache Hierarchy



Cache	P-core	E-core	LP E-core
L0 DL0	48KB 12-way set-associative per core	None	None
L1 DL1	192KB 12-way set-associative per core	32KB 8-way set-associative per core	32KB 8-way set-associative per core
L1 IL1	64KB 16-way set-associative per core	64KB 16-way set-associative per core	64KB 16-way set-associative per core
L2	3MB 12-way set-associative per core	4MB 16-way set-associative within a module of 4 Compute tile E-cores	2MB 16-way set-associative within a module of 2 SOC tile LP E-cores
L3	Maximum of 3 MB per P-core / module of 4 E-cores shared across Compute tile		None

12.7.4 Ring Interconnect

The Ring is a high speed, wide interconnect that links the processor cores, processor graphics and the System Agent.

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

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

12.7.5 Intel® Hybrid Technology

The processor contains two types of cores, denoted as big and small cores.

The big and small cores share the same instruction set and model specific registers (MSRs).

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

12.7.6 Intel® Hyper-Threading Technology

The processor supports Intel® Hyper-Threading Technology (Intel® HT Technology) that allows an execution processor IA core to function as two logical processors. While some execution resources such as caches, execution units, and buses are shared, each logical processor has its own architectural state with its own set of general-purpose registers and control registers. This feature should be enabled using the BIOS and requires operating system support.

NOTE

Intel® Hyper-Threading Technology is supported on U SKU only.

12.7.7 Intel® Turbo Boost Technology 2.0

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

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

NOTE

Intel® Turbo Boost Technology 2.0 is supported on U SKU when processor voltage and temperature allows.

12.7.7.1 Intel® Turbo Boost Technology 2.0 Power Monitoring

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

12.7.7.2 Intel® Turbo Boost Technology 2.0 Power Control

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

12.7.7.3 Intel® Turbo Boost Technology 2.0 Frequency

To determine the highest performance frequency amongst active processor IA cores, the processor takes the following into consideration:

- The number of processor IA cores operating in the C0 state.
- The estimated processor IA core current consumption and ICCMax settings.
- The estimated package prior and present power consumption and turbo power limits.
- The package temperature.

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

12.7.8 Intel® Turbo Boost Max Technology 3.0

The Intel® Turbo Boost Max Technology 3.0 (ITBMT 3.0) grants a different maximum Turbo frequency for individual processor cores.

To enable ITBMT 3.0 the processor exposes individual core capabilities; including diverse maximum turbo frequencies.

An operating system that allows for varied per core frequency capability can then maximize power savings and performance usage by assigning tasks to the faster cores, especially on low core count workloads.

Processors enabled with these capabilities can also allow software (most commonly a driver) to override the maximum per-core Turbo frequency limit and notify the operating system via an interrupt mechanism.

NOTE

Intel® Turbo Boost Max 3.0 Technology is supported on H SKUs when processor voltage and temperature allows. This technology is not supported on U SKUs.

12.7.9 Intel® Adaptive Boost Technology

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

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

12.7.10 Intel System Agent Enhanced SpeedStep ® Technology

Intel System Agent Enhanced SpeedStep ® Technology

Intel System Agent Enhanced SpeedStep ® Technology also known as SAGV (System Agent Geyserville) is a dynamic voltage frequency scaling of the System Agent clock based on memory utilization. Unlike processor core and package Enhanced Intel SpeedStep ® Technology, System Agent Enhanced SpeedStep ® Technology has three valid operating points. When running light workload and Intel System Agent Enhanced SpeedStep ® Technology is enabled, the DDR data rate may change as follows:

BIOS/MRC DDR training at maximum, mid and minimum frequencies sets I/O and timing parameters.

In order to achieve the optimal levels of performance and power, the memory initialization and training process performed during first system boot or after CMOS clear or after a BIOS update will take a longer time than a typical boot. During this initialization and training process, end users may see a blank screen. More information on the memory initialization process can be found in the industry standard JEDEC Specifications found on www.JEDEC.org.

Before changing the DDR data rate, the processor sets DDR to self-refresh and changes the needed parameters. The DDR voltage remains stable and unchanged.

12.7.11 Enhanced Intel SpeedStep® Technology

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

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

NOTE

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

12.7.12 Intel® Speed Shift Technology

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

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

Intel® Advanced Vector Extensions 2.0 (Intel® AVX2) is the latest expansion of the Intel instruction set. Intel® AVX2 extends the Intel® Advanced Vector Extensions (Intel® AVX) with 256-bit integer instructions, floating-point fused multiply-add (FMA) instructions, and gather operations. The 256-bit integer vectors benefit math, codec, image, and digital signal processing software. FMA improves performance in face detection, professional imaging, and high-performance computing. Gather operations increase vectorization opportunities for many applications. In addition to the vector extensions, this generation of Intel processors adds bit manipulation instructions useful in compression, encryption, and general purpose software. For more information on Intel® AVX, refer to http://www.intel.com/software/avx

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

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

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

12.7.13.1 Intel® AVX2 Vector Neural Network Instructions (AVX2 VNNI)

Vector instructions for deep learning extension for AVX2.

Similar functionality as the AVX-512 VNNI instruction set but limited to 256 bit AVX registers.

Unlike AVX-512 VNNI, this instruction set is available in hybrid computing.

12.7.14 Intel® 64 Architecture x2APIC

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

Specifically, x2APIC:

- Retains all key elements of compatibility to the xAPIC architecture:
  - Delivery modes

  - Interrupt and processor priorities
  - Interrupt sources
  - Interrupt destination types
- Provides extensions to scale processor addressability for both the logical and physical destination modes
- Adds new features to enhance the performance of interrupt delivery
- Reduces the complexity of logical destination mode interrupt delivery on link based architectures

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

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

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

12.7.15 Intel® Dynamic Tuning Technology (Intel® DTT)

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

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

12.7.16 Cache Line Write Back (CLWB)

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

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

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

12.7.17 User Mode Wait Instructions

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

TPAUSE is an enhanced *PAUSE* instruction.

The mnemonics for the three new instructions are:

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

12.8 Deprecated Technology

The processor has deprecated the following technology and it is no longer supported:

- DDR Running Average Power Limit (DDR RAPL)

12.9 Power and Internal Signals

12.9.1 Signal Description

Signal Name	Type	Description
GPP_V01/**ACPRESENT**	I	**ACPRESENT**: This input pin indicates when the platform is plugged into AC power or not. *Note:* An external pull-up resistor is required.
GPP_V00/**BATLOW#**	I	**Battery Low**: An input from the battery to indicate that there is insufficient power to boot the system. Assertion will prevent wake from S4/S5 states or exit from Deep Sx state. This signal can also be enabled to cause an SMI# when asserted. This signal is multiplexed with GPD0. *Note:* An external pull-up resistor is required.
PLT_PWROK	I	**PLT Power OK**: When asserted, is an indication to the PLT that all of its core power rails have been stable. The platform may drive asynchronously. When is de-asserted, the PLT asserts PLTRST#. *Notes:* • must not glitch, even if RSMRST# is low • An external pull-down resistor is required.
GPP_B13/**PLTRST#**/USB-C_GPP_B13	O	**Platform Reset**: The processor asserts PLTRST# to reset devices on the platform. The processor asserts PLTRST# low in Sx states and when a cold, warm, or global reset occurs. The processor de-asserts PLTRST# upon exit from Sx states and the aforementioned resets. There is no guaranteed minimum assertion time for PLTRST#.
GPP_A21/**PMCALERT#**	I/OD	**PMC Alert Pin**: Supports USB-C* PD controller architecture. *Note:* • An external pull-up resistor is required regardless of whether Integrated USB Type-C is used.
GPP_V03/**PWRBTN#**	I	**Power Button**: The Power Button may cause an SMI# or SCI to indicate a system request to go to a sleep state. If the system is already in a sleep state, this signal will cause a wake event. If PWRBTN# is pressed for more than 4 seconds (default; timing is configurable), this will cause an unconditional transition (power button override) to the S5 state. Override will occur even if the system is in the S4 states. This signal has an internal Pull-up resistor and has an internal 16 ms de-bounce on the input. *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 Reset**: This signal is used for resetting the primary power plane logic. This signal must be asserted for at least 10ms before de-asserting. *Note:* An external pull down resistor is required.
GPP_V06/**SLP_A#**	O	**SLP_A#**: Signal asserted when the Intel® CSME platform goes to M-Off or M3-PG. Depending on the platform, this pin may be used to control power to various devices that are part of the Intel® CSME sub-system in the platform. If you are not using SLP_A# for any functional purposes on your platform, or can tolerate lack of minimum assertion time, program the "SLP_A# minimum assertion width" value to the minimum. SLP_A# functionality can be utilized on the platform via either the physical pin or via the SLP_A# virtual wire over eSPI.
GPP_V12/**SLP_LAN#**	O	**(H/U only)** **LAN Sub-System Sleep Control**: When SLP_LAN# is de-asserted it indicates that the Platform LAN Connect Device must be powered. When SLP_LAN# is asserted, power can be shut off to the Platform LAN Connect Device. SLP_LAN# will always be de-asserted in S0 and anytime SLP_A# is de-asserted.

continued...

Signal Name	Type	Description
GPP_V09/**SLP_WLAN#**	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 will be de-asserted in S0. If you are not using SLP_WLAN# for any functional purposes on your platform, or can tolerate lack of minimum assertion time, program the "SLP_A# minimum assertion width" value to the minimum.
GPP_V04/**SLP_S3#**	O	**S3 Sleep Control:**SLP_S3# is for power plane control. This signal shuts off power to all non-critical systems when in the S4 or S5 state.
GPP_V05/**SLP_S4#**	O	**S4 Sleep Control**: SLP_S4# is for power plane control. This signal shuts power to all non-critical systems when in the S4 or S5 state. *Note:* This pin must be used to control the DRAM power in order to use the processor DRAM power-cycling feature.
GPP_V10/**SLP_S5#**	O	**S5 Sleep Control**: SLP_S5# is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 state.
GPP_V08/**SUSCLK**	O	**Suspend Clock**: This clock is a digitally buffered version of the RTC clock.
GPP_A02/ESPI_IO2/ **PRIMPWRDNACK**/USB-C_GPP_A02	O	**PRIMPWRDNACK**: Active high. Asserted by the processor on behalf of the Intel CSME when it does not require the processor Primary well to be powered.
GPP_F09/RSVD/ **SX_EXIT_HOLDOFF#**/ ISH_GP11/USB-C_GPP_F09	I	**Sx Exit Holdoff Delay**: Delay exit from Sx state after SLP_A# is de-asserted. *Note:* When eSPI is enabled, the flash sharing functionality using SX_EXIT_HOLDOFF# is not supported, but the pin still functions to hold off Sx exit after SLP_A# de-assertion.
SYS_RESET#	I	**System Reset**: This pin forces an internal reset after being de-bounced. *Note:* An external pull-up resistor is required.
GPP_B23/**TIME_SYNC1**/ ISH_GP6/USB-C_GPP_B23	I	**Time Synchronization**: Used for synchronization both input (latch time when pin asserted) and output (toggle pin when programmed time is hit).
GPP_E16/PROC_GP3/ **VRALERT#**/ISH_GP10/USB-C_GPP_E16	I	**VR Alert**: ICC Max throttling indicator from the processor voltage regulators. VRALERT# pin allows the VR to force processor throttling to prevent an over current shutdown. PMC based on the VRALERT# and messages from the processor. The messages from the processor allows the processor to constrain the processor to a particular power budget.
GPP_V14/**WAKE#**	I/OD	**PCI Express* Wake Event in Sx**: Input Pin in Sx. Sideband wake signal on PCI Express* asserted by components requesting wake up. *Notes:* • This is an output pin during S0ix states hence this pin cannot be used to wake up the system during S0ix states. • An external pull-up resistor is required.

12.9.2 Power Sequencing Signals

Table 39. Power Sequencing Signals

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
SKTOCC#	**Socket Occupied:** Pulled down directly in the processor package to the ground. System board designers may use this signal to determine if the processor is present for safety purposes, it helps to avoid accidentally applying power to the socket while nothing is installed into the socket.	N/A	N/A	SE	All Processor Series

continued...

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
	If the customers do not want to use or do not need to use the pin (PKG without socket), they can leave it floating.				
VIDSOUT	**VIDSOUT, VIDSCK, VIDALERT#:** These signals comprise a three-signal serial synchronous interface used to transfer power management information between the processor and the voltage regulator controllers.	I/O	I:GTL/ O:OD	SE	All Processor Series
VIDSCK		O	OD		
VIDALERT#		I	CMOS		

12.9.3 Integrated Pull-Ups and Pull-Downs

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

12.9.4 I/O Signal Planes and States

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

continued...

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

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

13.0 Power Delivery

13.1 Power and Ground Signals

Table 40. H and U - Processor Series Power Rail Description

Name	Description
VCCPRIM_1P8	Fixed 1.8 V for primary well.
VCCPRIM_3P3	Fixed 3.3 V for primary well.
VCCPRIM_IO	Fixed 1.25 V for IO blocks.
VCCPRIM_VNNAON	Fixed 0.77 V for digital core blocks.
VCCPRIM_VNNAON_FLTRA	VNNAON with filter requirements.
VCCPRIM_VNNAON_FLTRB	VNNAON with filter requirements.
VCCPRIM_1P8_FLTRA	VCCPRIM_1P8 with filter requirements.
VCCCORE	Dynamic SVID power rail for IA cores.
VCCGT	Dynamic SVID power rail for graphics.
VCCSA	Dynamic SVID power rail for system agent.
VDD2	Fixed 1.05/1.10 V power rail for memory host controller.
VCCRTC	RTC well supply. *Notes:* 1. VCCRTC nominal voltage is 1.5 V. This rail is intended to always come up first and always stay on. It should NOT be power cycled regularly on non-coin battery designs. 2. 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.
VSS	Ground

Table 41. H and U - Processor Series Power Rail Sense Signals

Name	Description
VCCCORE_SENSE	VCCCORE sense pin.
VCCGT_SENSE	VCCGT sense pin.
VCCSA_SENSE	VCCSA sense pin.
VCCPRIM_IO_SENSE	VCCPRIM_IO sense pin.
VCCPRIM_VNNAON_SENSE	VCCPRIM_VNNAON sense pin.
VCCCORE_VSS_SENSE	VCCCORE VSS sense pin.
VCCGT_VSS_SENSE	VCCGT VSS sense pin.
VCCSA_VSS_SENSE	VCCSA VSS sense pin.

13.2 Digital Linear Voltage Regulator (DLVR)

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

13.3 Fast V-Mode (FVM)

This power management feature insulates VR FETs / Inductors from observing full PL4 current as well as upstream input power devices from observing full PL4 power.

IccMAX.APP represents the real PL4 workload maximum expected current when FVM is enabled, which is less than IccMAX (PL4 current when FVM disabled).

Fast V-mode allows for platform power subsystems to be designed to IccMAX.APP , instead of IccMAX, while providing a performance improvement over proactive PL4 and IccMAX reduction.

Processor Series	VCC_{CORE}	VCC_{GT}	VCC_{SA}
H Processor 6P+8E 45W	Enabled	Enabled	Enabled
H Processor 6P+8E 28W	Enabled	Enabled	Enabled
U Processor 2P+8E 15W	Enabled	Disabled[1]	Enabled
Note: 1. VCC_{GT} FVM is disabled due to Itrip_max ≈ Iccmax, no added value to enable FVM.			

13.4 Current Excursion Protection (CEP)

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

IMVP9.2 VRs enhance the CEP detector by adding a cycle by cycle current limiting feature where the IMVPVR quickly enters cycle by cycle current limit (becomes a current source) with the VR output current limited to a preset value (ITRIP) as set in the ICC_limit register.

13.5 Reactive PL4 with PL4 Boost

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

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

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

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

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

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

14.0 Electrical Specifications

For information on Intel® Core™ Ultra Processor Electrical Specification, download the pdf, click on the navigation pane and refer the spreadsheet, **842704-001_U_H_Electrical_Specification.xlsx**.

15.0 Thermal Management

Table 42. Definitions/Acronyms

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

15.1 Processor Thermal Management

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

- Remains below the Maximum Operating Temperature (T_sensor_max) specification at the maximum Processor Base Power (PBP, previously known as TDP).
- Conforms to system constraints, such as system acoustics, system skin-temperatures, and exhaust-temperature requirements.

CAUTION

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

15.1.1 Thermal Considerations

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

The processor integrates multiple processing IA cores and graphics cores. This may result in power distribution differences across the package and should be considered when designing the thermal solution.

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

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

NOTE

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

15.1.1.1 Package Power Control

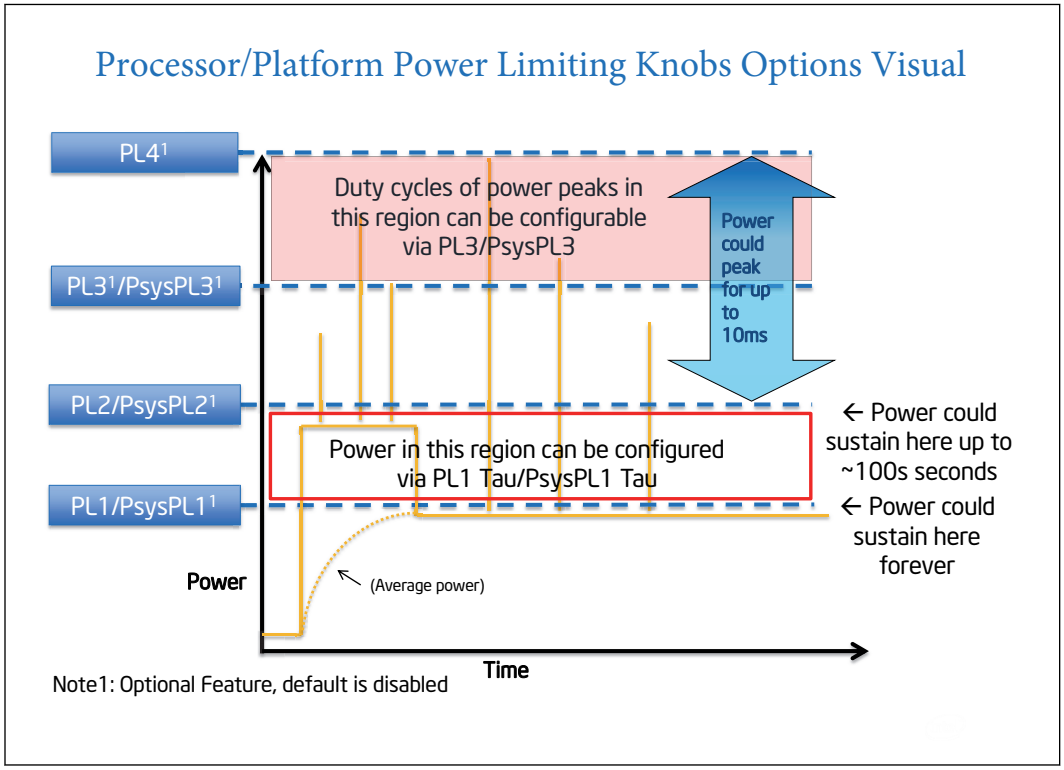
The package power control settings of PL1, PL2, PL3, PL4, and Tau allow the designer to configure Intel® Turbo Boost Technology 2.0 to match the platform power delivery and package thermal solution limitations.

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

NOTES

1. Implementation of Intel® Turbo Boost Technology 2.0 only requires configuring PL1, PL1, Tau and PL2.
2. PL3 is disabled by default.

Figure 14. Package Power Control



15.1.1.2 Platform Power Control

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

When the Psys signal is properly implemented, the system designer can utilize the package power control settings of PsysPL1/Tau, PsysPL2, and PsysPL3 for additional manageability to match the platform power delivery and platform thermal solution limitations for Intel® Turbo Boost Technology 2.0. The operation of the PsysPL1/tau, PsysPL2 and PsysPL3 are analogous to the processor power limits described in Package Power Control on page 132.

- **Platform Power Limit 1 (PsysPL1)**: A threshold for average platform power that will not be exceeded - recommend to set to equal platform thermal capability.
- **Platform Power Limit 2 (PsysPL2)**: A threshold that if exceeded, the PsysPL2 rapid power limiting algorithms will attempt to limit the spikes above PsysPL2.
- **Platform Power Limit 3 (PsysPL3)**: A threshold that if exceeded, the PsysPL3 rapid power limiting algorithms will attempt to limit the duty cycle of spikes above PsysPL3 by reactively limiting frequency.
- **PsysPL1 Tau**: An averaging constant used for PsysPL1 exponential weighted moving average (EWMA) power calculation.

- The Psys signal and associated power limits / Tau are optional for the system designer and disabled by default.
- The Psys data will not include power consumption for charging.
- The Intel Dynamic Tuning (DTT/DPTF) is recommended for performance improvement in mobile platforms. Dynamic Tuning is configured by system manufacturers dynamically optimizing the processor power based on the current platform thermal and power delivery conditions. Contact Intel Representatives for enabling details.

15.1.1.3 Turbo Time Parameter (Tau)

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

15.1.2 Thermal Management Features

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

15.1.2.1 Adaptive Thermal Monitor

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

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

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

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

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

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

Adaptive Thermal Monitor protection is always enabled.

TCC Activation Offset

TCC Activation Offset can be set as an offset from T_sensor_max to lower the onset of TCC and Adaptive Thermal Monitor. In addition, there is an optional time window (Tau) to manage processor performance at the TCC Activation offset value via an EWMA (Exponential Weighted Moving Average) of temperature.

TCC Activation Offset with Tau=0

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

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

TCC Activation Offset with Tau

To manage the processor with the EWMA (Exponential Weighted Moving Average) of temperature, an offset (degrees Celsius) is written to the TEMPERATURE_TARGET (1A2h) MSR, bits [29:24], and the time window (Tau) is written to the TEMPERATURE_TARGET (1A2h) MSR [6:0]. The Offset value will be subtracted from the value found in bits [23:16] and be the temperature.

The processor will manage to this average temperature by adjusting the frequency of the various domains. The instantaneous T_sensor can briefly exceed the average temperature. The magnitude and duration of the overshoot is managed by the time window value (Tau).

This averaged temperature thermal management mechanism is in addition, and not instead of T_sensor_max thermal management. That is, whether the TCC activation offset is 0 or not, TCC Activation will occur at T_sensor_max .

Frequency / Voltage Control

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

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

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

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

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

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

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

Clock Modulation

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

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

Thermal Throttling

As the processor approaches T_sensor_max a throttling mechanisms will engage to protect the processor from over-heating and provide control thermal budgets.

Achieving this is done by reducing IA and other subsystem agent's voltages and frequencies in a gradual and coordinated manner that varies depending on the dynamics of the situation. IA frequencies and voltages will be directed down as low as LFM (Lowest Frequency Mode), each E-core module (4 E-cores) or each P-core can be thermally throttle independently. Further restricts are possible via Thermal Threshold point (TT1) under conditions where thermal budget cannot be re-gained fast enough

with voltages and frequencies reduction alone. TT1 keeps the same processor voltage and clock frequencies the same yet skips clock edges to produce effectively slower clocking rates. This will effectively result in observed frequencies below LFM on the Windows PERF monitor.

15.1.2.2 Digital Thermal Sensor

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

Temperature values from the DTS can be retrieved through:

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

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

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

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

Fan Speed Control with Digital Thermal Sensor

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

15.1.2.3 PROCHOT# Signals

Intel recommends using PROCHOT# as an input signal to avoid Power, Thermal and Performance implications.

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

The PROCHOT# signal can be configured to the following modes:

- **Input Only**: PROCHOT is driven by an external device.
- **Output Only**: PROCHOT is driven by processor.
- **Bi-Directional**: Both Processor and external device can drive PROCHOT signal

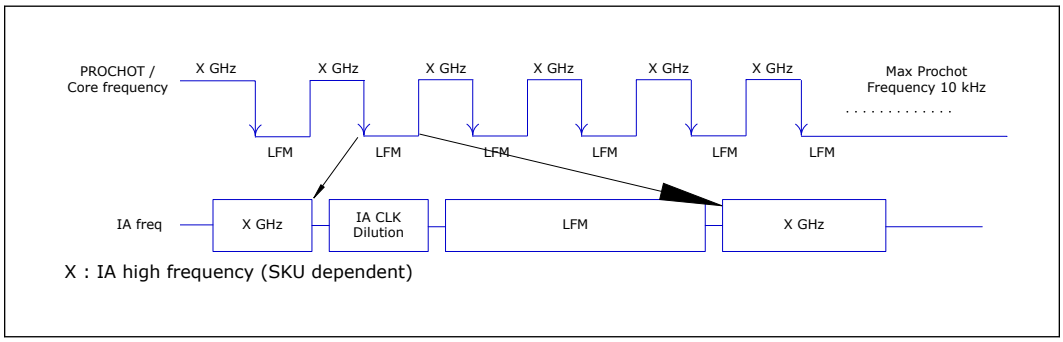
PROCHOT Input Only

The PROCHOT# signal should be set to input only by default. In this state, the processor will only monitor PROCHOT# assertions and respond by setting the maximum frequency to 10 khz.

The following two features are enabled when PROCHOT is set to Input only:

- **Fast PROCHOT**: Respond to PROCHOT# within 1us of PROCHOT# pin assertion, reducing the processor power.
- **PROCHOT Demotion Algorithm**: designed to improve system performance during multiple PROCHOT assertions.

Figure 15. PROCHOT Demotion Description



15.1.2.4 PROCHOT Output Only

Legacy state, PROCHOT is driven by the processor to external device.

15.1.2.5 Bi-Directional PROCHOT#

By default, the PROCHOT# signal is set to input only. When configured as an input or bi-directional signal, PROCHOT# can be used for thermally protecting other platform components should they overheat as well. When PROCHOT# is driven by an external device:

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

The processor package will remain at the lowest supported P-state until the system de-asserts PROCHOT#. The processor can be configured to generate an interrupt upon assertion and de-assertion of the PROCHOT# signal.

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

15.1.2.6 PROCHOT Demotion

PROCHOT Demotion is designed to improve system performance following multiple Platform PROCHOT consecutive assertions. During each PROCHOT assertion processor will eventually transition to the lowest P-State (Pn) supported by the processor IA cores and graphics cores (LFM). When detecting several PROCHOT consecutive assertions the processor will reduce the max frequency in order to reduce the PROCHOT assertions events. The processor will keep reducing the frequency until reaching LFM, the processor can further reduce the frequency using clock dilution (change in the duty cycle) until no PROCHOT consecutive assertions detected. The processor will keep reducing the frequency until no consecutive assertions detected. The processor will raise the frequency if no consecutive PROCHOT assertion events will occur. PROCHOT demotion algorithm enabled only when the PROCHOT is configured as input.

15.1.2.7 Voltage Regulator Protection using PROCHOT#

PROCHOT# may be used for thermal protection of voltage regulators (VR). System designers can create a circuit to monitor the VR temperature and assert PROCHOT# and, if enabled, activate the TCC when the temperature limit of the VR is reached. When PROCHOT# is configured as a bi-directional or input only signal, if the system assertion of PROCHOT# is recognized by the processor, results in power reduction. Power reduction down to LFM and duration of the platform PROCHOT# assertion. supported by the processor IA cores and graphics cores. Systems should still provide proper cooling for the VR and rely on bi-directional PROCHOT# only as a backup in case of system cooling failure. Overall, the system thermal design should allow the power delivery circuitry to operate within its temperature specification even while the processor is operating at its Adaptive Thermal Monitor protection is always enabled.

NOTE

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

15.1.2.8 Thermal Solution Design and PROCHOT# Behavior

With a properly designed and characterized thermal solution, it is anticipated that PROCHOT# will only be asserted for very short periods of time when running the most power intensive applications. The processor performance impact due to these brief

periods of TCC activation is expected to be so minor that it would be immeasurable. However, an under-designed thermal solution that is not able to prevent excessive assertion of PROCHOT# in the anticipated ambient environment may:

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

15.1.2.9 Low-Power States and PROCHOT# Behavior

Depending on package power levels during package C-states, outbound PROCHOT# may de-assert while the processor is idle as power is removed from the signal. Upon wake up, if the processor is still hot, the PROCHOT# will re-assert, although typically package idle state residency should resolve any thermal issues. The PECI interface is fully operational during all C-states and it is expected that the platform continues to manage processor IA core and package thermals even during idle states by regularly polling for thermal data over PECI.

15.1.2.10 THERMTRIP# Signal

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

15.1.2.11 Critical Temperature Detection

Critical Temperature detection is performed by monitoring the package temperature. This feature is intended for graceful shutdown before the THERMTRIP# is activated. However, the processor execution is not guaranteed between critical temperature and THERMTRIP#. If the Adaptive Thermal Monitor is triggered and the temperature remains high, a critical temperature status and sticky bit are latched in the PACKAGE_THERM_STATUS (1B1h) MSR and the condition also generates a thermal interrupt, if enabled.

15.1.2.12 Software Controlled Clock Modulation (On-Demand Mode)

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

15.1.3 Assured Power (cTDP)

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

NOTES

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

15.1.3.1 Assured Power (cTDP) Modes

NOTE

Assured Power availability may vary between the different SKUs.

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

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

Table 43. Assured Power

Mode	Description
Base	The average power dissipation and sensor temperature operating condition limit, is specified in Table 45 on page 144, Table 46 on page 145. For the SKU Segment and Configuration, for which the processor is validated during manufacturing when executing an associated Intel-specified high-complexity workload at the processor IA core frequency corresponding to the configuration and SKU.
Maximum Assured Power	The SKU-specific processor IA core frequency where manufacturing confirms logical functionality within the set of operating condition limits specified for the SKU segment and Maximum Assured Power (cTDP UP) configuration in Table 45 on page 144, Table 46 on page 145. The Maximum Assured Power (a.k.a. cTDP UP) Frequency and corresponding Processor Base Power is higher than the processor IA core Base Frequency and SKU Segment Base TDP.
Minimum Assured Power	The processor IA core frequency where manufacturing confirms logical functionality within the set of operating condition limits specified for the SKU segment and Configurable Minimum Assured Power (cTDP Down) configuration in Table 45 on page 144. The Minimum Assured Power (cTDP Down) Frequency and corresponding Processor Base Power is lower than the processor IA core Base Frequency and SKU Segment Processor Base Power.

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

15.1.3.2 Low Power Mode

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

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

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

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

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

15.1.4 Intel® Memory Thermal Management

DRAM Thermal Aggregation

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

DRAM Thermal Monitoring

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

DRAM Refresh Rate Control

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

DRAM Bandwidth Throttling (Change to DDR Bandwidth Throttling)

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

15.2 Processor Base Power Thermal and Power Specifications

Table 44. General Notes

This table is applicable to Table 45 on page 144, Table 46 on page 145, and Table 47 on page 145

Note	Definition
1	The Processor Base Power and Assured Power (cTDP) values are the average power dissipation in sensor temperature operating condition limit, for the SKU Segment and Configuration, for which the processor is validated during manufacturing when executing an associated Intel-specified high-complexity workload at the processor IA core frequency corresponding to the configuration and SKU.
2	Processor Base Power workload may consist of a combination of processor IA core intensive and graphics core intensive applications.
3	Can be modified at runtime by MSR writes, with MMIO and with PECI commands.
4	'Turbo Time Parameter' is a mathematical parameter (units of seconds) that controls the processor turbo algorithm using a moving average of energy usage. Do not set the Turbo Time Parameter to a value less than 0.1 seconds. refer to Platform Power Control on page 133 for further information.
5	The shown limit is a time averaged-power, based upon the Turbo Time Parameter. Absolute product power may exceed the set limits for short durations or under virus or uncharacterized workloads.
6	The Processor will be controlled to a specified power limit as described in Intel® Turbo Boost Technology 2.0 Power Monitoring on page 117. If the power value and/or 'Turbo Time Parameter' is changed during runtime, it may take a short time (approximately 3 to 5 times the 'Turbo Time Parameter') for the algorithm to settle at the new control limits.
7	This is a hardware default setting and not a behavioral characteristic of the part.
8	For controllable turbo workloads, the PL2 limit may be exceeded for up to 10ms.
9	LPM power level is an opportunistic power and is not a guaranteed value as usages and implementations may vary.
10	Power limits may vary depending on if the product supports the Minimum Assured Power (cTDP Down) and/or Maximum Assured Power (cTDP Up) modes. Default power limits can be found in the PKG_PWR_SKU MSR (614h).
11	The processor tile does not reach maximum sustained power simultaneously since the sum of all active circuit's estimated power budget is controlled to be equal to or less than the specified PL1 limit.
12	Minimum Assured Power(cTDP Down) is based on 128EU equivalent graphics configuration. Minimum Assured Power(cTDP Down) does not decrease the number of active Processor Graphics EUs but relies on Power Budget Management (PL1) to achieve the specified power level.
13	May vary based on SKU.

continued...

Note	Definition
14	• The formula of PL2=PL1*1.25 is the hardware. • PL2- Processor opportunistic higher Average Power with limited duration controlled by Tau_PL1 setting, the larger the Tau, the longer the PL2 duration.
15	Processor Base Power (TDP) workload does not reflect various I/O connectivity cases such as Thunderbolt.
16	Hardware default of PL1 Tau=1s, By including the benefits available from power and thermal management features the recommended is to use PL1 Tau=28s.
17	PL1 Tau max recommendation value is the default value in the BIOS/BKC and this value is been tested.

Table 45. Processor Base Power (TDP) Specifications (H-Processor)

Segment and Package	Processor P/E Cores, Graphics Configuration and Processor Base Power (TDP)	Configuration			Processor P/E Core Frequency [GHz]	Thermal Design Power (Processor Base Power (TDP)) [W]	General Notes
H-Processor BGA	6P+8E Core 45W	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	3.7 GHz	65	1,9,10,11,12, 15
				E-Core	3.2 GHz		
			Processor Base power (TDP)	P-Core	2.9 GHz	45	
				E-Core	2.7 GHz		
			Minimum Assured Power (cTDP Down)	P-Core	2.5 GHz	35	
				E-Core	1.9 GHz		
			Low Frequency Mode - LFM		0.4 GHz	N/A	
		Graphics Core Frequency	Graphics Frequency		0.8 GHz	N/A	
			Low Frequency Mode - LFM		0.1 GHz		
H-Processor BGA	6P+8E Core 28W	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	3.7 GHz	65	1,9,10,11,12, 15
				E-Core	3.2 GHz		
			Processor Base power (TDP)	P-Core	2GHz up to 2.2GHz	28	
				E-Core	1.5GHz up to 1.7GHz		
			Minimum Assured Power (cTDP Down)	P-Core	1.2 GHz	20	
				E-Core	0.9 GHz		
			Low Frequency Mode - LFM		0.4 GHz	N/A	
		Graphics Core Frequency	Graphics Frequency		0.8 GHz	N/A	
			Low Frequency Mode - LFM		0.1 GHz		
H-Processor BGA	4P+8E Core 28W	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	3.7GHz up to 4GHz	65	1,9,10,11,12, 15
				E-Core	3.2GHz up to 3.5GHz		
			Processor Base power (TDP)	P-Core	1.7GHz up to 2.4GHz	28	
				E-Core	1.3GHz up to 1.8GHz		

continued...

Segment and Package	Processor P/E Cores, Graphics Configuration and Processor Base Power (TDP)	Configuration			Processor P/E Core Frequency [GHz]	Thermal Design Power (Processor Base Power (TDP)) [W]	General Notes
			Minimum Assured Power (cTDP Down)	P-Core	1.2GHz up to 1.3GHz	20	
				E-Core	0.9GHz up to 1GHz		
			Low Frequency Mode - LFM		0.4 GHz	N/A	
		Graphics Core Frequency	Graphics Frequency		0.8 GHz		
			Low Frequency Mode - LFM		0.1 GHz		

Refer to General Notes

Table 46. Processor Base Power Specifications (U-Processor)

Segment and Package	Processor P/E Cores, Graphics Configuration and Processor Base Power (TDP)	Configuration			Processor P/E Core Frequency	Thermal Design Power (Processor Base Power (TDP)) [W]	General Notes
U-Processor BGA	2P+8E Core 15W	IA Core Frequency	Maximum Assured Power (cTDP Up)	P-Core	3.1GHz up to 3.4GHz	28	1,9,10,11,12, 15
				E-Core	2.6GHz up to 2.8GHz		
			Processor Base power (TDP)	P-Core	1.5GHz up to 2.1GHz	15	
				E-Core	1.3GHz up to 1.7GHz		
			Minimum Assured Power (cTDP Down)	P-Core	1.2GHz up to 1.8GHz	12	
				E-Core	0.9GHz up to 1.5GHz		
			Low Frequency Mode - LFM		0.4 GHz	N/A	
		Graphics Core Frequency	Graphics Frequency		0.8 GHz		
			Low Frequency Mode - LFM		0.1 GHz		

Refer to General Notes

15.3 Processor Thermal and Power Specifications

Table 47. Package Turbo Specifications (H/U -Processors)

No Specifications for Min/Max PL1/PL2 values.

Hardware default of PL1 Tau=1s, By including the benefits available from power and thermal management features the recommended is to use PL1 Tau=28s for less than 45W. For 45W the recommended is to use PL1 Tau=56s

PL2- Processor opportunistic higher Average Power – Reactive, Limited Duration controlled by Tau_PL1 setting.

PL1 Tau - PL1 average power is controlled via PID algorithm with this Tau, The larger the Tau, the longer the PL2 duration.

System cooling solution and designs found to not being able to support the Performance TauPL1, adjust the TauPL1 to cooling capability.

Segment and Package	Processor IA Cores, Graphics, Configuration and Processor Base Power	Parameter	Minimum	Tau MSR Max Value	Recommended Value	Units	General Notes
H-Processor	6P+8E Core 45W	Power Limit 1 Time (PL1 Tau)	0.1	448	56	S	3,4,5,6,7,8,14,16,17
		Power Limit 1 (PL1)	N/A	N/A	45	W	
		Power Limit 2 (PL2)	N/A	N/A	115	W	
H-Processor	6P+8E Core 28W	Power Limit 1 Time (PL1 Tau)	0.1	448	28	S	3,4,5,6,7,8,14,16,17
		Power Limit 1 (PL1)	N/A	N/A	28	W	
		Power Limit 2 (PL2)	N/A	N/A	60	W	
U-Processor	2P+8E Core 15W	Power Limit 1 Time (PL1 Tau)	0.1	448	28	S	3,4,5,6,7,8,14,16,17
		Power Limit 1 (PL1)	N/A	N/A	15	W	
		Power Limit 2 (PL2)	N/A	N/A	57	W	
H-Processor	4P+8E Core 28W	Power Limit 1 Time (PL1 Tau)	0.1	448	28	S	3,4,5,6,7,8,14,16,17
		Power Limit 1 (PL1)	N/A	N/A	9	W	
		Power Limit 2 (PL2)	N/A	N/A	60	W	

Note: Refer to General Notes

Table 48. Processor Temperature Specifications (H/U- Processors)

Segment	Symbol	Package Turbo Parameter	Temperature Range		Processor Base Power Specification Temperature Range		Units	Notes
			Minimum	Maximum	Minimum	Maximum		
H - Processor BGA	T_sensor	Sensor temperature limit	0	105	35	105	ºC	1, 3
U - Processor BGA	T_sensor	Sensor temperature limit	0	105	35	105	ºC	1, 2, 3

Notes:
1. The thermal solution needs to ensure that the processor temperature does not exceed the Processor temperature range.
2. Thermal designs, if desired, can program a TCC Offset and Tau value to limit the processors operational T_sensor.
3. The processor sensor temperature is monitored by Digital Temperature Sensors (DTS). For DTS accuracy, refer to Digital Thermal Sensor on page 137

15.4 Error and Thermal Protection Signals

Table 49. Error and Thermal Protection Signals

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
CATERR#	**Catastrophic Error:** This signal indicates that the system has experienced a catastrophic error and cannot continue to operate. The processor will set this signal for non-recoverable machine check errors or other unrecoverable internal errors. CATERR# is used for signaling the following types of errors: Legacy MCERRs, CATERR# is asserted for 16 BCLKs. Legacy IERRs, CATERR# remains asserted until warm or cold reset.	O	OD	SE	All Processor Series
PECI	**Platform Environment Control Interface:** A serial sideband interface to the processor. It is used primarily for thermal, power, and error management.	I/O	PECI, Async	SE	All Processor Series
PROCHOT#	**Processor Hot:** PROCHOT# goes active when the processor temperature monitoring sensor(s) detects that the processor has reached its maximum safe operating temperature. This indicates that the processor Thermal Control Circuit (TCC) has been activated, if enabled. This signal can also be driven to the processor to activate the TCC.	I/O	I:GTL/ O:OD	SE	All Processor Series
THERMTRIP#	**Thermal Trip:** The processor protects itself from catastrophic overheating by use of an internal thermal sensor. This sensor is set well above the normal operating temperature to ensure that there are no false trips. The processor will stop all executions when the operating temperature exceeds approximately 125 °C. This is signaled to the system by the THERMTRIP# pin.	O	OD	SE	All Processor Series

15.5 Thermal Sensor

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

15.5.1 Modes of Operation

The Thermal sensors have two usages when enabled:

1. One use is to provide the temperature of the Processor in units of 1 ºC. There is a 8 bit field for the temperature, with a theoretical range from –128 ºC to +127 ºC. Practically the operational range for the system is between -40 ºC and 125 ºC.
2. The second use is to allow programmed trip points to cause alerts to SW or in the extreme case shutdown. Temperature may be provided without having any SW alerts set.

There are two thermal alert capabilities. One is for the catastrophic event (thermal runaway) which results in an immediate system power down (S5 state). The other alert provides an indication to the platform that a particular temperature has been caused. This second alert needs to be routed to SMI or SCI based on SW programming.

15.5.2 Temperature Trip Point

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

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

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

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

15.5.3 Thermal Sensor Accuracy (Taccuracy)

The processor 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.
- No accuracy is specified for temperature range beyond 110 °C or below -10 °C.

15.5.4 Thermal Reporting to EC

To support a platform EC that is managing the system thermals, the processor provides the ability for the EC to read the processor temperature over SMBus and/or over eSPI. If enabled, Power Management will drive the temperature directly to the SMBus and eSPI units. The EC will issue an SMBus read or eSPI OOB Channel request and receives a single byte of data, indicating a temperature between 0°C and 127°C, where 255 (0xFF) indicates that the sensor isn't enabled yet. The EC must be connected to either SMLink1 or eSPI for thermal reporting support.

15.5.5 Thermal Trip Signal (SOCHOT#)

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

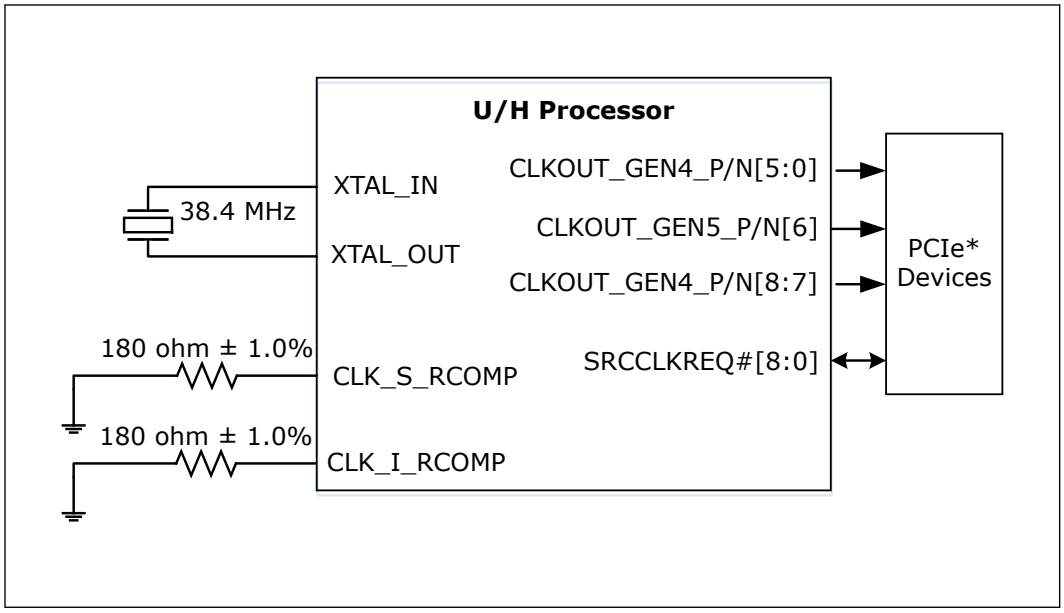
SOCHOT# is an O/D output and requires a Pull-up on the motherboard.

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

16.0 System Clocks

16.1 Integrated Clock Controller (ICC)

Figure 16. ICC Diagram



16.1.1 Signal Description

Table 50. Signal Description

Signal Name	Type	SSC Capable	Description	Availability
CLKOUT_GEN4_N[5:0] **CLKOUT_GEN4_P[5:0]** **CLKOUT_GEN5_N6** **CLKOUT_GEN5_P6** **CLKOUT_GEN4_N[8:7]** **CLKOUT_GEN4_P[8:7]**	O	Yes	**PCI Express* Clock Output**: Serial Reference 100 MHz PCIe* specification compliant differential output clocks to PCIe* devices	U/H
GPP_D04/**IMGCLKOUT0**/USB-C_GPP_D04 GPP_D00/**IMGCLKOUT1**/USB-C_GPP_D00 GPP_F07/RSVD/**IMGCLKOUT2**/USB-C_GPP_F07 GPP_F08/RSVD/**IMGCLKOUT3**/USB-C_GPP_F08	O		**Imaging Clock** : Clock for external camera sensor.	U/H

continued...

Signal Name	Type	SSC Capable	Description	Availability
GPP_D07/**IMGCLKOUT4**/ ISH_UART0_RTS#/ISH_SPI_MISO/ USB-C_GPP_D07				
GPP_C09/**SRCCLKREQ0#**/USB-C_GPP_C09 GPP_C10/**SRCCLKREQ1#**/USB-C_GPP_C10 GPP_C11/**SRCCLKREQ2#**/USB-C_GPP_C11 GPP_C12/**SRCCLKREQ3#**/USB-C_GPP_C12 GPP_C13/**SRCCLKREQ4#**/USB-C_GPP_C13 GPP_D21/RSVD/**SRCCLKREQ5#**/ USB-C_GPP_D21	IOD		**Clock Request**: Serial Reference Clock request signals for PCIe* 100 MHz differential clocks	U/H
GPP_D18/**SRCCLKREQ6#**/USB-C_GPP_D18 GPP_D19/**SRCCLKREQ7#**/USB-C_GPP_D19 GPP_D20/**SRCCLKREQ8#**/USB-C_GPP_D20				U/H
XTAL_IN	I		**Crystal Input:** Input connection for 38.4 MHz crystal to Processor	U/H
XTAL_OUT	O		**Crystal Output**: Output connection for 38.4 MHz crystal to Processor	U/H
CLK_S_RCOMP **CLK_I_RCOMP**	Analog		**Differential Clock Bias Reference**: Used to set BIAS reference for differential clocks.	U/H

Notes: 1. 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.

2. **U-Series Processor:**
   a. SRCCLKREQ#[5:0] signals can be configured to map to any of the PCIe Lanes 1-12 while using clock output differential pairs CLKOUT_GEN4_P/N[5:0].
   b. SRCCLKREQ#[8:6] signals can be configured to map to any of the PCIe Lanes 13-20 while using clock output differential pairs CLKOUT_GEN5_P/N[6] or CLKOUT_GEN4_P/N[8:7].
3. **H-Series Processor**:
   a. SRCCLKREQ#[5:0] signals can be configured to map to any of the PCIe Lanes 1-12 while using clock output differential pairs CLKOUT_GEN4_P/N[5:0].
   b. SRCCLKREQ#[8:6] signals can be configured to map to any of the PCIe Lanes 13-28 while using clock output differential pairs CLKOUT_GEN5_P/N[6] or CLKOUT_GEN4_P/N[8:7].
   c. **Applicable to Gen5 PCIe Devices only**: SRCCLKREQ#[8:6] signals can be configured to map to any of the PCIe Lanes 21-28 while using clock output differential pair CLKOUT_GEN5_P/N[6].

16.2 I/O Signal Pin States

Table 51. I/O Signal Pin States

Signal Name	Power Plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CLKOUT_GEN4/5_P[0:8]	Primary	Toggling	Toggling	OFF (Gated Low)

continued...

Signal Name	Power Plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CLKOUT_GEN4/5_N[0:8]				
SRCCLKREQ[0:8]#	Primary	Un-driven	Un-driven	Un-driven
1. Reset reference for primary well pins is RSMRST#.				

16.3 Clock Topology

The processor has three reference clocks that drive the various components within the processor:

- PCIe reference clock (PCTGLK). 100 MHz with SSC.
- Fixed clock. 38.4 MHz without SSC (crystal clock).

PCTGLK drives the following clock domains:

- PCIe Controller(s)

Fixed clock drives the following clock domains:

- Display
- Serial Voltage Identification (SVID) controller
- Time Stamp Counters (TSC)
- USB Type-C* subsystem

16.3.1 Integrated Reference Clock PLL

The processor includes a phase lock loop (PLL) that generates the reference clock for the processor from a fixed crystal clock. The processor reference clock is also referred to as Base Clock or BCLK.

The BCLK PLL has controls for RFI/EMI mitigations as well as Overclocking capabilities.

17.0 Real Time Clock (RTC)

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

The real-time clock performs two key functions:

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

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

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

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

Table 52. Acronyms

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

17.1 Signal Description

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

continued...

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

17.2 I/O Signal Planes and States

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

18.0 Memory

18.1 System Memory Interface

18.1.1 Processor SKU Support Matrix

Table 53. DDR Support Matrix Table

Technology	DDR5	LPDDR5[9]	LPDDR5x [9]	LPCAMM2
Processor	**H/U**	**H/U**	**H/U**	**H/U**
Maximum Frequency [MT/s]	1R/2R - 6400[10]	6400 [10]	Type 3: 1R/2R - 7467 [10] Type 4: 1R/2R - 8400 [10]	LPDDR5x Type3 - 7467 [10]
VDDQ [V] [6]	1.1	0.52	0.52	0.52
VDD2 [V] [6]	1.1	1.065	1.065	1.065
DPC [1]	1	-	-	-
Maximum RPC [2]	2	2	2	2
Die Density [Gb]	16, 24, 32	8,12,16	8,12,16	16
Ballmap Mode	NIL	NIL	NIL	NIL

Notes:
1. 1DPC refer to when only 1DIMM slot per channel is routed.
2. RPC = Rank Per Channel
3. An Interleave SoDIMM/MD placements like butterfly or back-to-back supported with a Non-Interleave ballmap mode at H/U Processor
4. Memory down of all technologies should be implemented homogeneous means that all DRAM devices should be from the same vendor and have the same part number. Implementing a mix of DRAM devices may cause serious signal integrity and functional issues.
5. There is no support for memory modules with different technologies or capacities on opposite sides of the same memory module. If one side of a memory module is populated, the other side is either identical or empty.
6. DDR5: VDD2 is Processor voltage, VDDQ is DRAM voltage merged with VDD2.
7. Pending DRAM samples availability and eco system readiness.
8. 5V is DIMM voltage, 1.1V is Memory down voltage
9. LPDDR5 technology supports 8 Bank Mode, BG (Bank Group) Mode and 16 Bank Mode.LPDDR5x technology supports BG Mode and 16 Bank Mode, according to JEDEC spec. Bank Mode may vary according to SAGV Point.
10. DDR POR speed refers to Processor top SKU. Other SKUs may use lower memory speed, refer to ark.intel.com for top memory speed .

18.1.2 Supported Memory Modules and Devices

Table 54. Supported DDR5 Non-ECC SoDIMM/CSoDIMM Module Configurations (H/U-Processor Series)

Raw Card Version	DIMM Capacity [GB]	DRAM Device Technology [Gb]	DRAM Organization	# of DRAM Devices	# of Ranks	# of Row/Col Address Bits	# of Banks Inside DRAM	Page Size [K]
A , F	16	16	2048M x 8	8	1	17/10	16	8
C , H	8	16	1024M x 16	4	1	17/10	8	8
B , G	32	16	2048M x 8	16	2	17/10	16	8
A , F	24	24	3072M x 8	8	1	17/10	32	8
C , H	12	24	1536M x 16	4	1	17/10	16	8
B , G	48	24	3072M x 8	16	2	17/10	32	8
A , F	32	32	4096M x 8	8	1	17/10	32	8
C , H	16	32	2048M x 16	4	1	17/10	16	8
B , G	64	32	4096M x 8	16	2	17/10	32	8

Table 55. Supported DDR5 Memory Down Device Configurations (H/U - Processor Series)

Maximum System Capacity [GB][2]	PKG Type (Die bits x Package bits)	DRAM Organization / Package Type	Package Density [Gb]	Die Density [Gb]	Dies Per Channel	Rank Per Channel	PKGs Per channel	Physical Device Rank	Banks Inside DRAM	Page Size [K]
32	SDP 8x8	2048M x 8	16	16	8	1	8	1	16	8
16	SDP 16x16	1024M x 16	16	16	4	1	4	1	8	8
48	SDP 8x8	3072M x 8	24	24	8	1	8	1	32	8
24	SDP 16x16	1536M x 16	24	24	4	1	4	1	16	8
64	SDP 8x8	4096M x 8	32	32	8	1	8	1	32	8
32	SDP 16x16	2048M x 16	32	32	4	1	4	1	16	8

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

Table 56. Supported LPDDR5/x x32 DRAMs Configurations (H/U - Processor Series)

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

continued...

Maximum System Capacity [GB][3]	PKG Type [2]	(Die bits per Ch x PKG bits)	Die Density [Gb]	PKG Density [Gb]	Rank Per PKGs
16	QDP	16x32	8	32	2
32	ODP	16x32	8	64	2

Notes: 1. x32 BGA devices are 315 balls
2. DDP - Dual Die Package, QDP - Quad Die Package, ODP - Octal Die Package
3. Maximum system capacity refers to system with all 8 sub-channels populated

Table 57. Supported LPDDR5/x x64 DRAMs Configurations (H/U - Processor Series)

Maximum System Capacity [GB][2]	PKG Type	(Die bits per Ch x PKG bits)[2]	Die Density [Gb]	PKG Density [Gb]	DRAM Channels Per PKGs	Rank Per PKGs
16[1]	QDP	16x64	16	64	4	1
32[1]	ODP	16x64	16	128	4	2
8[1]	QDP	16x64	8	32	4	1
16[1]	ODP	16x64	8	64	4	2

Notes: 1. QDP = Quad Die Package, ODP-Octal Die Package
2. Maximum system capacity refers to system with all 8 sub-channels populated

18.1.3 System Memory Timing Support

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

- tCL = CAS Latency
- tRCD = Activate Command to READ or WRITE Command delay
- tRP = PRECHARGE Command Period
- tRPb = per-bank PRECHARGE time
- tRPab = all-bank PRECHARGE time
- CWL = CAS Write Latency
- Command Signal modes:
  - 2N indicates a new DDR5 command may be issued every 2 clocks
  - 1N indicates a new DDR5/LPDDR5/x command may be issued every clock

Table 58. DDR System Memory Timing Support

DRAM Device	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRP (ns)	CWL (tCK)	CMD Mode
DDR5	4800	40	16.00	16.00	38	2N
DDR5	5600	46	16.00	16.00	44	2N
DDR5	6400	52	16	16	50	2N

Table 59. LPDDR System Memory Timing Support

DRAM Device	Transfer Rate (MT/s)	tCL (tCK)	tRCD (ns)	tRPpb (ns)	tRPab (ns)	WL (tCK) Set B
LPDDR5	6400	17	18	18	21	16
LPDDR5x	7466	20	18	18	21	19
LPDDR5x	8400	23	18	18	21	22

18.1.3.1 SAGV Points

SAGV (System Agent Geyserville) is a way by which the SOC can dynamically scale the work point (V/F), by applying DVFS (Dynamic Voltage Frequency Scaling) based on memory bandwidth utilization and/or the latency requirement of the various workloads for better energy efficiency at System-Agent. Pcode heuristics are in charge of providing request for Qclock work points by periodically evaluating the utilization of the memory and IA stalls.

Table 60. SA Speed Enhanced Speed Steps (SA-GV) and Gear Mode Frequencies

Processor	Technology	DDR Maximum Rate [MT/s]	SAGV-LowBW	SAGV-MedBW	SAGV-HighBW	SAGV- High Performance
H/U	LPDDR5x	8400	3200 G4	4800 G4	6400 G4	8400 G4
	LPDDR5x	7466	3200 G4	4800 G4	6400 G4	7466 G4
U	DDR5	6400	3200 G4	4800 G4	6400 G4	5600 G2
	DDR5	5600	3200 G4	4800 G4	5200 G4	5600 G2
H	DDR5	6400	3200 G4	4800 G4	6000 G4	6400 G2
	DDR5	5600	3200 G4	4800 G4	5200 G4	5600 G2

Notes: 1. Intel® Core Ultra 200H and 200U Series Processors supports dynamic gearing technology where the Memory Controller can run at 1:2 (Gear-2 mode) or 1:4 (Gear-4 mode) ratio of DRAM speed. The gear ratio is the ratio of DRAM speed to Memory Controller Clock .

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

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

DDR Frequency Shifting

DDR interfaces emit electromagnetic radiation which can couple to the antennas of various radios that are integrated in the system, and cause radio frequency interference (RFI). The DDR Radio Frequency Interference Mitigation (DDR RFIM) feature is primarily aimed at resolving narrowband RFI from DDR5 and LPDDR5/x technologies for the Wi-Fi* high and ultra-high bands (~5-7 GHz) . By changing the DDR data rate, the harmonics of the clock can be shifted out of a radio band of interest, thus mitigating RFI to that radio. This feature is working with SAGV on, one of the SAGV points is used as RFI mitigation point.

18.1.4 Memory Controller (MC)

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

The two controllers are independent and have no means of communicating with each other, they need to be configured separately.

In a symmetric memory population, each controller provides access to half of the total physical memory address space.

18.1.5 System Memory Controller Organization Mode

The IMC supports two memory organization modes, single-channel and dual-channel. Depending upon how the DDR Schema and DIMM Modules are populated in each memory channel, a number of different configurations can be applied.

Single-Channel Mode

In this mode, all memory accesses are directed to a single Memory Controller. Single-Channel mode is used when either the MC0 or MC1 are populated, but not both MC are populated.

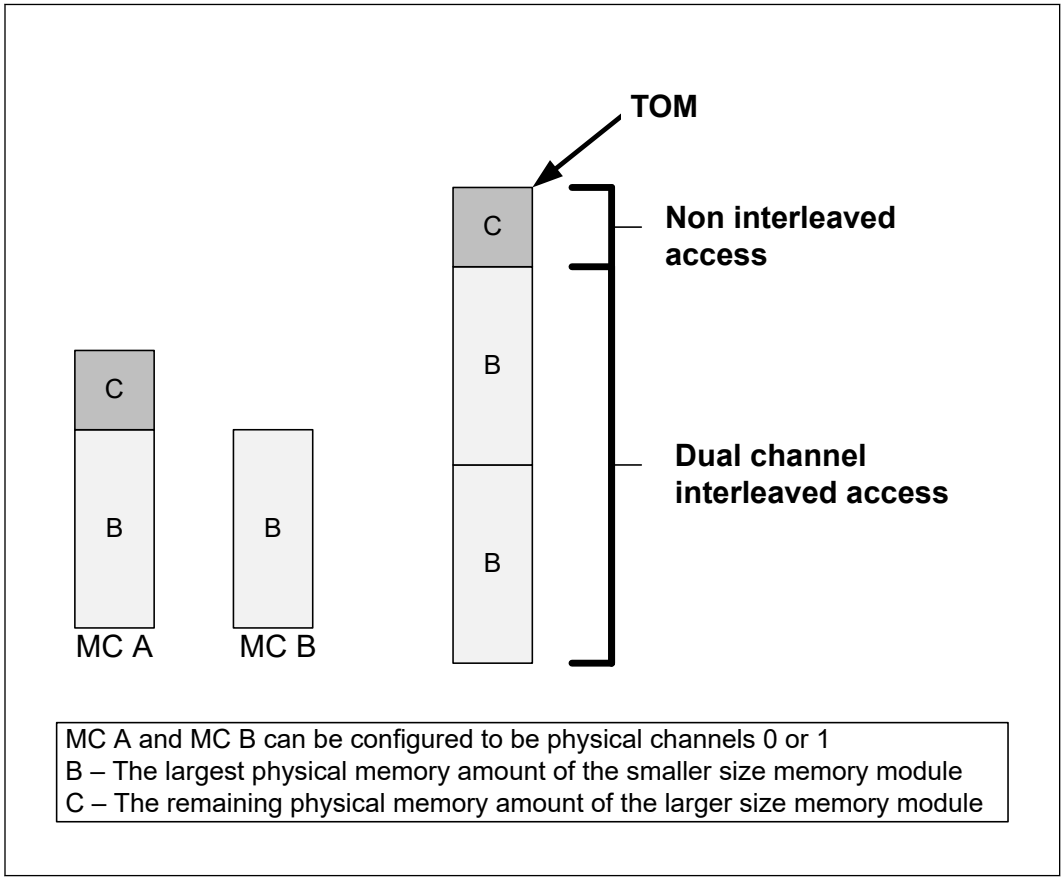
Dual-Channel Mode – Intel® Flex Memory Technology Mode (DDR5 Only)

The IMC supports Intel Flex Memory Technology Mode. Memory is divided into a symmetric and asymmetric zones. The symmetric zone starts at the lowest address in each MC and is contiguous until the asymmetric zone begins or until the top address of the channel with the smaller capacity is reached. In this mode, the system runs with one zone of dual-channel mode and one zone of single-channel mode, simultaneously, across the whole memory array.

NOTE

MC A and MC B are mapped for physical MC0 and MC1 respectively or vice versa; however, Channel A size should be greater or equal to Channel B size, BIOS is mapping MC A and MC B after identifying populated capacity in each channel .

Figure 17. Intel® DDR5 Flex Memory Technology Operations



Dual-Channel Symmetric Mode (Interleaved Mode)

Dual-Channel Symmetric mode, also known as interleaved mode, this mode used when both MC0 and MC1 are populated with same memory capacity. This mode provides maximum performance on real world applications compared to Single channel and Flex Memory Technology mode. Addresses are toggled between the channels. If there are two requests, and the second request is to an address on the opposite channel from the first, that request can be sent before data from the first request has returned. If two consecutive cache lines are requested, both may be retrieved simultaneously. .

When both MCs are populated with the same memory capacity and the boundary between the dual channel zone and the single channel zone is the top of memory, IMC operates completely in Dual-Channel Symmetric mode.

NOTES

- The DDR5 DRAM device technology, die density, rank and width may vary from one channel to another.
- Memory Organization Mode depend on memory capacity only.
- Different memory size between channels are relevant to DDR5 only.

18.1.6 System Memory Frequency

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

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

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

Just-in-Time Command Scheduling

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

Command Overlap

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

Out-of-Order Scheduling

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

18.1.8 Data Scrambling

The system memory controller incorporates a Data Scrambling feature to minimize the impact of excessive di/dt on the platform system memory VRs due to successive 1s and 0s on the data bus. Past experience has demonstrated that traffic on the data bus is not random and can have energy concentrated at specific spectral harmonics creating high di/dt which is generally limited by data patterns that excite resonance between the package inductance and on die capacitances. As a result, the system memory controller uses a data scrambling feature to create pseudo-random patterns on the system memory data bus to reduce the impact of any excessive di/dt.

18.1.9 Data Swapping

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

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

NOTE

All DRAM devices sharing ZQ resistor must be connected to the same MC channel.

18.1.10 LPDDR5/x CMD/ADD Ascending and Descending

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

Table 61. LPDDR5/x CMD/ADD Ascending and Descending

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

NOTE

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

18.1.11 DDR I/O Interleaving

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

18.1.12 DRAM Clock Generation

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

18.1.13 DRAM Reference Voltage Generation

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

18.1.14 Data Swizzling

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

18.1.15 Error Correction With Standard RAM

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

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

18.1.16 Post Package Repair (PPR)

PPR is supported according to JEDEC Spec.

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

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

18.2 Integrated Memory Controller (IMC) Power Management

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

18.2.1 Disabling Unused System Memory Outputs

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

- Reduced power consumption.
- Reduced possible overshoot/undershoot signal quality issues seen by the processor I/O buffer receivers caused by reflections from potentially unterminated transmission lines.

When a given rank is not populated, the corresponding control signals (CLK_P/CLK_N/CS) are not driven.

At reset, all rows should be assumed to be populated, until it can be proven that they are not populated. This is due to the fact that when CS is tri-stated with a DRAMs present, the DRAMs are not ensured to maintain data integrity. CS tri-state should be enabled by BIOS where appropriate, since at reset all rows should be assumed to be populated.

18.2.2 DRAM Power Management and Initialization

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

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

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

The different power-down modes supported are:

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

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

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

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

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

18.2.2.1 Conditional Self-Refresh

During S0 idle state, system memory may be conditionally placed into self-refresh state when the processor is in package C3 or deeper power state. Refer to Intel® Rapid Memory Power Management (Intel® RMPM) on page 109 for more details on conditional self-refresh with Intel® HD Graphics enabled.

When entering the S3 – Suspend-to-RAM (STR) state or S0 conditional self-refresh, the processor IA core flushes pending cycles and then enters SDRAM ranks that are not used by the processor graphics into self-refresh. The CKE signals remain LOW so the SDRAM devices perform self-refresh.

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

18.2.2.2 Dynamic Power-Down

Dynamic power-down of memory is employed during normal operation. Based on idle conditions, a given memory rank may be powered down. The IMC implements aggressive CKE control to dynamically put the DRAM devices in a power-down state.

The processor IA core controller can be configured to put the devices in active power down or pre-charge power-down. Pre-charge power-down provides greater power savings but has a bigger performance impact, since all pages will first be closed before putting the devices in power-down mode.

If dynamic power-down is enabled, all ranks are powered up before doing a refresh cycle and all ranks are powered down at the end of the refresh.

18.2.2.3 DRAM I/O Power Management

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

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

18.2.3 DDR Electrical Power Gating

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

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

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

In S3 or C-state transitions, the DDR does not go through training mode and will restore the previous training information.

18.2.4 Power Training

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

18.3 Signal Description

Table 62. DDR5 Memory Interface

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR0_DQ[3:0][7:0] DDR1_DQ[3:0][7:0] DDR2_DQ[3:0][7:0] DDR3_DQ[3:0][7:0]	**Data Buses**: Data signals interface to the SDRAM data buses. **Example:** DDR0_DQ2[5] refers to DDR channel 0, Byte 2, Bit 5.	I/O	DDR5	SE	H/U Processor
DDR0_DQSP[3:0] DDR0_DQSN[3:0] DDR1_DQSP[3:0] DDR1_DQSN[3:0] DDR2_DQSP[3:0] DDR2_DQSN[3:0] DDR3_DQSP[3:0] DDR3_DQSN[3:0]	**Data Strobes:** Differential data strobe pairs. The data is captured at the crossing point of DQS during reading and write transactions. **Example:** DDR0_DQSP0 refers to DQSP of DDR channel 0, Byte 0.	I/O	DDR5	Diff	H/U Processor
DDR0_CLK[1:0]_P DDR0_CLK[1:0]_N DDR1_CLK[1:0]_P DDR1_CLK[1:0]_N DDR2_CLK[1:0]_P DDR2_CLK[1:0]_N DDR3_CLK[1:0]_P DDR3_CLK[1:0]_N	**SDRAM Differential Clock:** Differential clocks signal pairs, pair per rank. The crossing of the positive edge and the negative edge of their complement are used to sample the command and control signals on the SDRAM.	O	DDR5	Diff	H/U Processor
DDR0_CS[1:0] DDR1_CS[1:0] DDR2_CS[1:0] DDR3_CS[1:0]	**Chip Select**: (1 per rank). These signals are used to select particular SDRAM components during the active state. There is one Chip Select for each SDRAM rank. The Chip select signal is Active Low.	O	DDR5	SE	H/U Processor
DDR0_CA[12:0] DDR1_CA[12:0] DDR2_CA[12:0] DDR3_CA[12:0]	**Command Address**: These signals are used to provide the multiplexed command and address to the SDRAM.	O	DDR5	SE	H/U Processor
DDR_RCOMP	**System Memory Resistance Compensation**	A	A	SE	H/U Processor
DRAM_RESET#	**Memory Reset**	O	CMOS	SE	H/U Processor

Table 63. LPDDR5/x Memory Interface

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR0_DQ[1:0][7:0] DDR1_DQ[1:0][7:0] DDR2_DQ[1:0][7:0] DDR3_DQ[1:0][7:0] DDR4_DQ[1:0][7:0] DDR5_DQ[1:0][7:0] DDR6_DQ[1:0][7:0]	**Data Buses**: Data signals interface to the SDRAM data buses. **Example**: DDR0_DQ[1][5] refers to DDR channel 0, Byte 1, Bit 5.	I/O	LPDDR5/x	SE	All Processor Series

continued...

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR7_DQ[1:0][7:0]					
DDR0_DQSP[1:0] DDR1_DQSP[1:0] DDR2_DQSP[1:0] DDR3_DQSP[1:0] DDR4_DQSP[1:0] DDR5_DQSP[1:0] DDR6_DQSP[1:0] DDR7_DQSP[1:0] DDR0_DQSN[1:0] DDR1_DQSN[1:0] DDR2_DQSN[1:0] DDR3_DQSN[1:0] DDR4_DQSN[1:0] DDR5_DQSN[1:0] DDR6_DQSN[1:0] DDR7_DQSN[1:0]	**Data Strobes**: Differential data strobe pairs. The data is captured at the crossing point of DQS during reading and write transactions.	I/O	LPDDR5/x	Diff	All Processor Series
DDR0_CLK_N DDR0_CLK_P DDR1_CLK_N DDR1_CLK_P DDR2_CLK_N DDR2_CLK_P DDR3_CLK_N DDR3_CLK_P DDR4_CLK_N DDR4_CLK_P DDR5_CLK_N DDR5_CLK_P DDR6_CLK_N DDR6_CLK_P DDR7_CLK_N DDR7_CLK_P	**SDRAM Differential Clock**: Differential clocks signal pairs, pair per channel and package. The crossing of the positive edge and the negative edge of their complement are used to sample the command and control signals on the SDRAM.	O	LPDDR5/x	Diff	All Processor Series
DDR0_CS[1:0] DDR1_CS[1:0] DDR2_CS[1:0] DDR3_CS[1:0] DDR4_CS[1:0] DDR5_CS[1:0] DDR6_CS[1:0] DDR7_CS[1:0]	**Chip Select**: (1 per rank). These signals are used to select particular SDRAM components during the active state. There is one Chip Select for each SDRAM rank. The Chip select signal is Active High.	O	LPDDR5/x	SE	All Processor Series
DDR0_CA[6:0] DDR1_CA[6:0] DDR2_CA[6:0] DDR3_CA[6:0] DDR4_CA[6:0] DDR5_CA[6:0] DDR6_CA[6:0] DDR7_CA[6:0]	**Command Address**: These signals are used to provide the multiplexed command and address to the SDRAM.	O	LPDDR5/x	SE	All Processor Series

continued...

Signal Name	Description	Dir.	Buffer Type	Link Type	Availability
DDR0_WCK_P DDR0_WCK_N DDR1_WCK_P DDR1_WCK_N DDR2_WCK_P DDR2_WCK_N DDR3_WCK_P DDR3_WCK_N DDR4_WCK_P DDR4_WCK_N DDR5_WCK_P DDR5_WCK_N DDR6_WCK_P DDR6_WCK_N DDR7_WCK_P DDR7_WCK_N	**Write Clock**s: WCK_N and WCK_P are differential clocks used for WRITE data capture and READ data output.	O	LPDDR5/x	Diff	All Processor Series
DDR_RCOMP	**System Memory Resistance Compensation**	A	A	SE	All Processor Series
DRAM_RESET#	**Memory Reset**	O	CMOS	SE	All Processor Series

19.0 USB Type-C* Sub System

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

The USB Type-C sub-system supports USB 3.2, USB4*, DPoC (DisplayPort over Type-C) protocols. The USB Type-C sub-system can also be configured as native DisplayPort 1.4a/2.1 or HDMI 2.1 interfaces, for more information refer to Display on page 204.

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

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

NOTE

If USB4 (20 Gb/s) only solutions are implemented, Thunderbolt 3 compatibility as defined by USB4/USB-PD specs and 15 W of bus power are still recommended.

19.1 General Capabilities

- xHCI (USB 3.2 host controller) and xDCI (USB 3.2 Gen 1x1 device controller) implemented in the processor.
- Intel® AMT/vPro over Thunderbolt™ docking.
- Support power saving when USB Type-C* disconnected.
- Support up to four simultaneous ports.
- DbC Enhancement for Low Power Debug until Pkg C6
- Host
  - Aggregate BW through the controller at least 3 GB/s, direct connection or over USB4.
  - Wake capable on each host port from S0i3, Sx.
- Device
  - Aggregate BW through xDCI controller at max 5 GB/s
  - D0i2 and D0i3 power gating
  - Wake capable on host initiated wakes when the system is in S0i3, Sx Available on all ports.
- Port Routing Control for Dual Role Capability
  - Needs to support SW/FW and ID pin based control to detect host versus device attach.

— SW mode requires PD controller or other FW to control.

- USB-R device to host controller connection is over UTMI+ links.

Table 64. USB Type-C* Port Configuration

	Port	U/H IOE-P-Series Processor
Group A	TCP 0	USB4[4] USB 3.2[3] DisplayPort[1] HDMI[2]
	TCP 1	
Group B	TCP 2	
	TCP 3	

Notes:
1. Supported on Type-C or Native connector (Fixed DP up to HBR3 link rate)
2. Supported only on Native connector.
3. USB 3.2 supported link rates:
   a. USB 3.2 Gen 1x1 (5 Gbps)
   b. USB 3.2 Gen 2x1 (10 Gbps)
   c. USB 3.2 Gen 2x2 (20 Gbps)
4. USB4 operating link rates (including both rounded and non-rounded modes for Thunderbolt™ 3 compatibility):
   a. USB4 Gen 2x2 (20 Gbps)
   b. USB4 Gen 3x2 (40 Gbps)
   c. 10.3125 Gbps, 20.625 Gbps per lane - Compatible to Thunderbolt™ 3 non-rounded modes.
5. USB 2.0 interface supported over Type-C connector.
6. Port group is defined as two ports sharing USB4 router, each router supports up to two display interfaces.
7. Display interface can be connected directly to a DP/HDMI/Type-C port or through USB4 router on a Type-C connector.
8. If two ports in the same group are configured to one as USB4 and the other as DP/HDMI fixed connection each port will support single display interface.

Table 65. USB Type-C* Lanes Configuration

Lane1	Lane2	Comments
USB4 / TBT3	USB4 / TBT3	Both lanes operate at same speed, one of (20.6 Gbps/10.3 Gbps/20 Gbps/10 Gbps)
USB4 / TBT3	No connect	20.6g/10.3g/20g/10g
No connect	USB4 / TBT3	
USB 3.2	USB 3.2	Multi-Lane USB 3.2 (Host Only), 2x10G = 20G
USB 3.2	No connect	Any combination of: USB 3.2 Gen 1x1 (5Gb/s) USB 3.2 Gen 2x1 (10Gb/s)
No connect	USB 3.2	
USB 3.2	DPx2	Any of HBR3/HBR2/HBR1/HRBR for DP1.4a, **DP2.1 (2x10/20 Gbps)**, and USB 3.2 (10 Gbps)
DPx2	USB 3.2	
DPx4	Both lanes at same DP rate - no support for 2x DPx2 USB Type-C connector	Any of HBR3/HBR2/HBR1/HRBR for DP1.4a, **DP2.1 (4x10/20 Gbps)**

Table 66. USB Type-C* Non-Supported Lane Configuration

Lane1	Lane2	Comments
-	PCIe* Gen3/2/1	No PCIe* native support
PCIe* Gen3/2/1	-	
-	USB4 / TBT3	No support for USB4 / TBT3 with any other protocol
USB4 / TBT3	-	

19.2 USB4* Router

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

USB4 architecture (formerly known as Thunderbolt™ 3 protocol) is a transformational high-speed, dual protocol I/O, and it provides flexibility and simplicity by encapsulating both data (PCIe* & USB 3.2) and video

(DisplayPort*) on a single cable connection that can daisy-chain up to five devices. USB4/Thunderbolt™ controllers act as a point of entry or a point of exit in the USB4 domain. The USB4 domain is built as a daisy chain of USB4/Thunderbolt™ enabled products for the encapsulated protocols - PCIe, USB 3.2 and DisplayPort. These protocols are encapsulated into the USB4 fabric and can be tunneled across the domain.

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

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

19.2.1 USB4 Host Router Implementation Capabilities

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

- Up to two DisplayPort* sink interfaces each one capable of:
  - DisplayPort 1.4 specification for tunneling
  - 1.62 Gbps or 2.7 Gbps or 5.4 Gbps or 8.1 Gbps link rates
  - x1, x2 or x4 lane operation
  - Support for DSC compression
- Up to two PCI Express* Root Port interfaces each one capable of:
  - PCI Express* 3.0 x4 compliant @ 8.0 GT/s
- Up to two xHCI Port interfaces each one capable of:
  - USB 3.2 Gen 2x1 (10 Gbps)
  - USB 3.2 Gen 2x2 (20 Gbps)
- USB4 Host Interface:

  - PCI Express* 3.0 x4 compliant endpoint
  - Supports simultaneous transmit and receive on 12 paths
  - Raw mode and frame mode operation configurable on a per-path basis
  - MSI and MSI-X support
  - Interrupt moderation support
- USB4 Time Management Unit (TMU):
- Up to two Interfaces to USB Type-C* connectors, each one supports:
  - USB4 PD entry mode, as well as TBT 3 compatibility mode, each supporting:
    - 20 paths per port
    - Each port support 20.625/20.0 Gbps or 10.3125/10.0 Gbps link rates per lane.
    - 16 counters per port

19.3 xHCI/xDCI Controllers

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

19.3.1 USB 3 Controllers

19.3.1.1 Extensible Host Controller Interface (xHCI)

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

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

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

19.3.1.2 Extensible Device Controller Interface (xDCI)

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

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

The xDCI controller support link rate of up to USB 3.2 Gen 1x1 (5G).

NOTE

These controllers are instantiated in the processor as a separate PCI function functionality for the USB-C* capable ports.

19.3.2 PCIe Interface

Table 67. PCIe via USB4 Configuration

USB4 IPs	USB4_PCIe	U/H IOE-P Processor Line USB Type-C* Ports
USB4_DMA0	USB4_PCIE0	TCP0
	USB4_PCIE1	TCP1
USB4_DMA1	USB4_PCIE2	TCP2
	USB4_PCIE3	TCP3

19.4 Display Interface

Refer to Display on page 204.

19.5 USB Type-C Signals

Signal Name	Description	Dir.	Link Type	Availability
TCP[1:0]_TX[1:0]_P TCP[1:0]_TX[1:0]_N	TX Data Lane.	O	Diff	H/U-Series Processor
TCP[3:2]_TX[1:0]_P TCP[3:2]_TX[1:0]_N	TX Data Lane.	O	Diff	H/U-Series Processor
TCP[1:0]_TXRX[1:0]_P TCP[1:0]_TXRX[1:0]_N	RX Data Lane, also serves as the secondary TX data lane.	I/O	Diff	H/U-Series Processor
TCP[3:2]_TXRX[1:0]_P TCP[3:2]_TXRX[1:0]_N	RX Data Lane, also serves as the secondary TX data lane.	I/O	Diff	H/U-Series Processor
TCP[1:0]_AUX_P TCP[1:0]_AUX_N	Common Lane AUX-PAD.	I/O	Diff	H/U-Series Processor
TCP[3:2]_AUX_P TCP[3:2]_AUX_N	Common Lane AUX-PAD.	I/O	Diff	H/U-Series Processor
TCP_RCOMP	Type-C Resistance Compensation.	A		H/U-Series Processor

19.6 AUX BIAS Control

On processor which support integrated USB Type-C* subsystem, the AUX BIAS control is required on the USB Type-C implementation (without retimer) for orientation connections. The functionality is muxed with certain GPIO pins. Refers to the GPIO implementation document for more information on the muxing and supported GPIO pin on the specific platform. In order to use the GPIO pin correctly for AUX BIAS control, the correct native functionality need to be configured and the correct Virtual Wire Index bit position need to be programmed in the BIOS policy.

Figure 18. GPIO - Virtual Wire Index Bit Mapping

GPIO Pin Group	Virtual Wire Index	Bit Position*
USB-C_GPP_[A06:A00]	10h	[6h:0h]
USB-C_GPP_[A15:11]	11h	[7h:3h]
USB-C_GPP_[A20:A16]	12h	[4h:0h]
USB-C_GPP_[B07:B00]	10h	[7h:0h]
USB-C_GPP_[B15:B08]	11h	[7h:0h]
USB-C_GPP_[B23:B16]	12h	[7h:0h]
USB-C_GPP_[C07:C00]	10h	[7h:0h]
USB-C_GPP_[C13:C08]	11h	[5h:0h]
USB-C_GPP_[C15]	11h	[7h]
USB-C_GPP_[C23:C16]	12h	[7h:0h]
USB-C_GPP_[D07:D00]	13h	[7h:0h]
USB-C_GPP_[D15:D08]	14h	[7h:0h]
USB-C_GPP_[D23:D16]	15h	[7h:0h]
USB-C_GPP_[E02:E00]	12h	[7h:5h]
USB-C_GPP_[E10:E03]	13h	[7h:0h]
USB-C_GPP_[E17:E11]	14h	[6h:0h]
USB-C_GPP_[E22]	15h	[3h]
USB-C_GPP_[F07:F00]	19h	[7h:0h]
USB-C_GPP_[F15:F08]	1Ah	[7h:0h]
USB-C_GPP_[F23:F16]	1Bh	[7h:0h]
USB-C_GPP_[H02:H00]	16h	[2h:0h]
USB-C_GPP_[H07:H04]	16h	[7h:4h]
USB-C_GPP_[H11:H08]	17h	[3h:0h]
USB-C_GPP_[H15:H13]	17h	[7h:5h]
USB-C_GPP_[H17:H16]	18h	[1h:0h]
USB-C_GPP_[H22:H19]	18h	[6h:3h]

NOTE

1. The bit position corresponds to each corresponding GPIO pin in the group.

   For example: the bit position for USB-C_GPP_A0 is bit 0h in Virtual Wire Index 10h.

20.0 Universal Serial Bus (USB)

The processor implements an xHCI USB 3.2 controller which provides support for up to 10 USB 2.0 signal pairs and 2 USB 3.2 signal pairs. The xHCI controller supports wake up from sleep states S1-S4. The xHCI controller supports up to 64 devices for a maximum number of 2048 Asynchronous endpoints (Control / Bulk) or maximum number of 128 Periodic endpoints (Interrupt / isochronous).

Table 68. Acronyms

Acronyms	Description
xHCI	eXtensible Host Controller Interface

20.1 Functional Description

20.1.1 eXtensible Host Controller Interface (xHCI) Controller

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

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

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

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

20.2 Signal Description

Signal Name	Type	Description	Availability
USB32_1_RXN **USB32_1_RXP**	I	**USB 3.2 Differential Receive Pair 1**: These are USB 3.2-based high-speed differential signals for Port 1. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB32_1_TXN **USB32_1_TXP**	O	**USB 3.2 Differential Transmit Pair 1**: These are USB 3.2-based high-speed differential signals for Port 1. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB32_2_RXN **USB32_2_RXP**	I	**USB 3.2 Differential Receive Pair 2**: These are USB 3.2-based high-speed differential signals for Port 2. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB32_2_TXN **USB32_2_TXP**	O	**USB 3.2 Differential Transmit Pair 2**: These are USB 3.2-based high-speed differential signals for Port 2. The signal should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_1 **USB2N_1**	I/O	**USB 2.0 Port 1 Transmit/Receive Differential Pair 1:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_2 **USB2N_2**	I/O	**USB 2.0 Port 2 Transmit/Receive Differential Pair 2:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_3 **USB2N_3**	I/O	**USB 2.0 Port 3Transmit/Receive Differential Pair 3:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_4 **USB2N_4**	I/O	**USB 2.0 Port 4 Transmit/Receive Differential Pair 4:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_5 **USB2N_5**	I/O	**USB 2.0 Port 5 Transmit/Receive Differential Pair 5:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_6 **USB2N_6**	I/O	**USB 2.0 Port 6 Transmit/Receive Differential Pair 6:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_7 **USB2N_7**	I/O	**USB 2.0 Port 7 Transmit/Receive Differential Pair 7:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor

continued...

Signal Name	Type	Description	Availability
USB2P_8 **USB2N_8**	I/O	**USB 2.0 Port 8 Transmit/Receive Differential Pair 8:** This USB 2.0 signal pair are routed to xHCI controller and should be mapped to a USB connector with one of the OC (overcurrent) signals.	H/U-Series Processor
USB2P_9 **USB2N_9**	I/O	**USB 2.0 Port 9 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 OC (overcurrent) signals.	H/U-Series Processor
USB2P_10 **USB2N_10**	I/O	**USB 2.0 Port 10 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 OC (overcurrent) signals.	H/U-Series Processor
GPP_E09/**USB_OC0#**/USB-C_GPP_E09	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred. When configured as OC# pin, a 10 kohm pull-up resistor is required to be connected to the power-rail. When this pin is configured as GPIO, no pull-up resistor is required. *Notes:* 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	H/U-Series Processor
GPP_B11/**USB_OC1#**/ DDSP_HPD2/DISP_MISC3/ USB-C_GPP_B11	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred. When configured as OC# pin, a 10 kohm pull-up resistor is required to be connected to the power-rail. When this pin is configured as GPIO, no pull-up resistor is required. *Notes:* 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	H/U-Series Processor
GPP_B14/**USB_OC2#**/ DDSP_HPD3/DISP_MISC4/ USB-C_GPP_B14	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred. When configured as OC# pin, a 100 kohm pull-up resistor is required to be connected to the power-rail. When this pin is configured as GPIO, no pull-up resistor is required. *Notes:* 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	H/U-Series Processor
GPP_B15/**USB_OC3#**/USB-C_GPP_B15	I	**Overcurrent Indicators**: This signal set the corresponding bit in the xHCI controller to indicate that an overcurrent condition has occurred.	H/U-Series Processor

continued...

Signal Name	Type	Description	Availability
		When configured as OC# pin, a 10 kohm pull-up resistor is required to be connected to the power-rail. When this pin is configured as GPIO, no pull-up resistor is required. *Notes:* 1. OC# pins are not 5 V tolerant. 2. OC# pins can be shared between USB ports. 3. Each USB connector should only have one OC# pin protection.	
USB2_1_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	H/U-Series Processor
USB2_2_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	H/U-Series Processor
USB32_RCOMP	A	USB Resistor Bias, analog connection points for an external resistor 200 ohm ± 1% connected to GND.	H/U-Series Processor

20.3 Integrated Pull-Ups and Pull-Downs

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

20.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[2]	Immediately After Reset[2]	S4/S5
USB32_[2:1]_RXN **USB32_[2:1]_RXP**	Primary	Internal Pull-Down	Internal Pull-Down	Internal Pull-Down
USB32_[2:1]_TXN **USB32_[2:1]_TXP**	Primary	Internal Pull-Down	Internal Pull-Down	Internal Pull-Down
USB2N_[10:1]	DSW	Internal Pull-Down	Internal Pull-Down	Internal Pull-Down
USB2P_[10:1]	DSW	Internal Pull-Down	Internal Pull-Down	Internal Pull-Down
USB_OC0#	Primary	Undriven	Undriven	Undriven
USB_OC1#	Primary	Undriven	Undriven	Undriven
USB_OC2#	Primary	Undriven	Undriven	Undriven
USB_OC3#	Primary	Undriven	Undriven	Undriven
USB2_[2:1]_RCOMP	Primary	Undriven	Undriven	Undriven
USB32_RCOMP	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

20.5 Supported USB 2.0 Ports

Due to the USB 2.0 port requirement for integrated Bluetooth® functionality with the integrated Intel® Wireless-AX (CNVi) solution, the following USB port will be available:

Figure 19. Supported USB 2.0 Ports on H-Processor

PROCESSOR SKU	Max USB 2.0 Nbr of Ports	USB 2.0 P1	USB 2.0 P2	USB 2.0 P3	USB 2.0 P4	USB 2.0 P5	USB 2.0 P6	USB 2.0 P7	USB 2.0 P8	USB 2.0 P9	USB 2.0 P10 (or Intel® Wireless-AX)	USBr1	USBr2
H/U-Processor Lines	10												

		Port Enabled

21.0 Intel® Volume Management Device (Intel® VMD) Technology

Objective

Standard Operating Systems generally recognize individual PCIe Devices and load individual drivers. This is undesirable in some cases such as, for example, when there are several PCIe-based hard-drives connected to a platform where the user wishes to configure them as part of a RAID array. The Operating System current treats individual hard-drives as separate volumes and not part of a single volume.

In other words, the Operating System requires multiple PCIe devices to have multiple driver instances, making volume management across multiple host bus adapters (HBAs) and driver instances difficult.

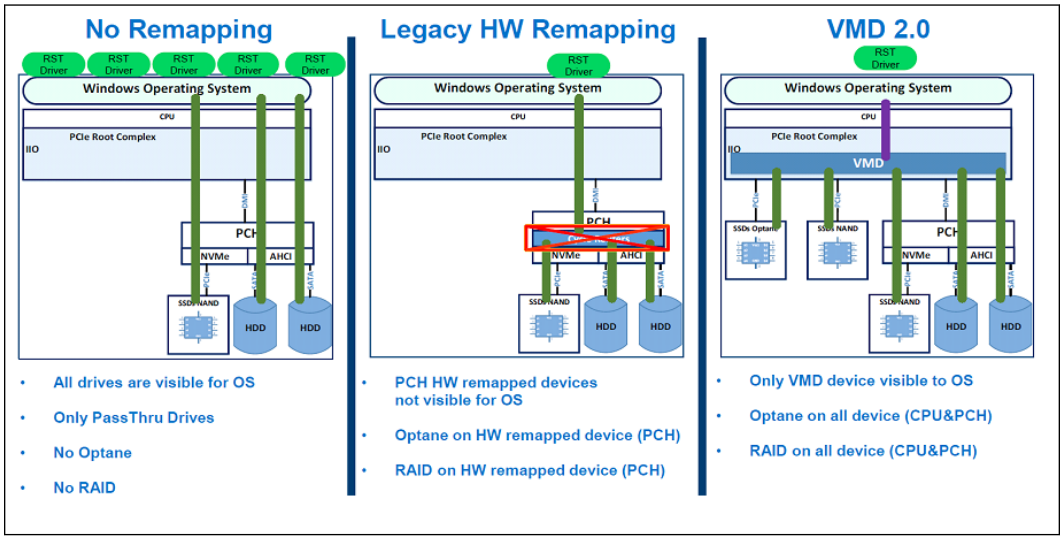
Intel® Volume Management Device (Intel® VMD) technology provides a means to provide volume management across separate PCI Express HBAs and SSDs without requiring operating system support or communication between drivers. For example, the OS will see a single RAID volume instead of multiple storage volumes, when Volume Management Device is used.

Technology Description

Intel® Volume Management Device technology does this by obscuring each storage controller from the OS, while allowing a single driver to be loaded that would control each storage controller.

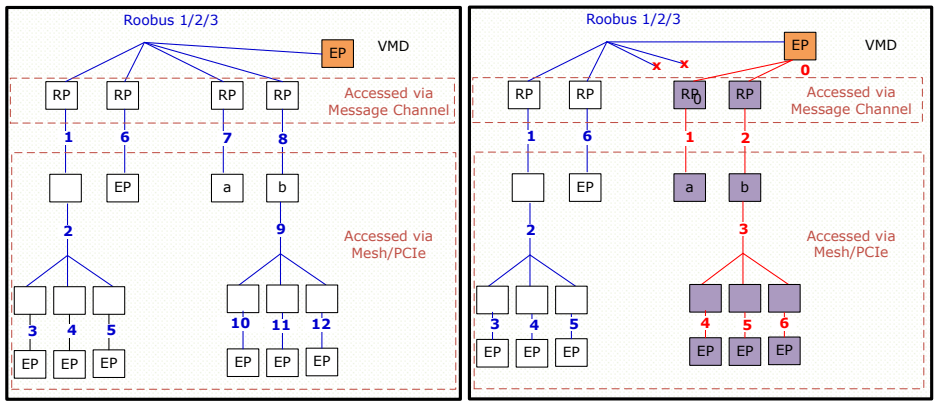
Intel® Volume Management technology requires support in BIOS and driver, memory and configuration space management.

Figure 20. Technology Description



A Volume Management Device (VMD) exposes a single device to the operating system, which will load a single storage driver. The VMD resides in the processor's PCIe root complex and it appears to the OS as a root bus integrated endpoint. In the processor, the VMD is in a central location to manipulate access to storage devices which may be attached directly to the processor or indirectly. Instead of allowing individual storage devices to be detected by the OS and therefore causing the OS to load a separate driver instance for each, VMD provides configuration settings to allow specific devices and root ports on the root bus to be invisible to the OS.

Access to these hidden target devices is provided by the VMD to the single, unified driver.



Features Supported

Supports MMIO mapped Configuration Space (CFGBAR):

- Supports MMIO Low
- Supports MMIO High
- Supports Register Lock or Restricted Access
- Supports Device Assign
- Function Assign
- MSI Remapping Disable

22.0 PCI Express* (PCIe*)

Table 69. Acronym

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

Table 70. Reference Table

Specification	Location
PCI Express® Base Specification Revision 5.0 Version 1.0, 22 May 2019	https://pcisig.com/
PCI Express M.2 Specification Revision 4.0, Version 1.1, April 14, 2022	https://pcisig.com/
PCI Express® Card Electromechanical Specification, Revision 5.0, Version 1.0, June 9, 2021	https://pcisig.com/

22.1 Functional Description

Table 71. Features Supported

PCIe Controller Feature	Processor-H Controllers					
	Processor-U Controllers					6
	1	2	3	4	5	
L1 Sub-States (L1.0, L1.1, L1.2)	Yes	Yes	Yes	Yes	Yes	Yes
L0s Link State (RX/TX)	Yes	Yes	Yes	Yes	Yes	Yes
S4/S5 Sleep States (Sx)	Yes	Yes	Yes	Yes	Yes	Yes
Common Clock Mode	Yes	Yes	Yes	Yes	Yes	Yes
Separate Reference Clock with Independent SSC (SRIS)	Yes	Yes	Yes	Yes	Yes	No
Separate Reference Clock with No SSC (SRNS)	Yes	Yes	Yes	Yes	Yes	No
Precision Time Management (PTM)	Yes	Yes	Yes	Yes	Yes	Yes
Advanced Error Reporting (AER)	Yes	Yes	Yes	Yes	Yes	Yes
End-to-End Lane Reversal	Yes	Yes	Yes	Yes	Yes	Yes
Latency Tolerance Reporting (LTR)	Yes	Yes	Yes	Yes	Yes	Yes
PCIe TX Half Swing	No	No	No	No	No	No
PCIe TX Full Swing	Yes	Yes	Yes	Yes	Yes	Yes
Run Time D3 (RTD3)	Yes	Yes	Yes	Yes	Yes	Yes
RTD3 through PFET_EN	Yes	Yes	Yes	Yes	Yes	Yes

continued...

PCIe Controller Feature	Processor-H Controllers					
	Processor-U Controllers					6
	1	2	3	4	5	
Access Control Services (ACS)	Yes	Yes	Yes	Yes	Yes	Yes
Alternative Routing-ID Interpretation (ARI)	Yes	Yes	Yes	Yes	Yes	Yes
Port 80h Decode	Yes	Yes	Yes	Yes	Yes	Yes
Lane Polarity Inversion	Yes	Yes	Yes	Yes	Yes	Yes
PCIe Controller Root Port Hot-Plug	Yes	Yes	Yes	Yes	Yes	Yes
Downstream Port Containment (DPC)	No	No	No	No	No	No
Enhanced Downstream Port Containment (eDPC)	No	No	No	No	No	No
Virtual Channel (VC)	0	0	0/1	0/1	0/1	0
NVMe Cycle Router	No	No	No	No	No	No
Volume Management Device (Intel® VMD)	Yes	Yes	Yes	Yes	Yes	Yes
RAID[0] and RAID[1] Mode Support [1,2]	Yes	Yes	Yes	Yes	Yes	Yes
RAID[5] Mode Support [3]	No	No	No	No	No	No
RAID[10] Mode Support [3]	No	No	No	No	No	No
Mammoth Glacier Discrete Device Support (M.2 1px2, 1px4)	Yes	Yes	Yes	Yes	Yes	Yes
Hybrid Dual Port Module Support (M.2 2px2)	Yes	Yes	No	No	No	No
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) Mem Write Transactions	RPs between PC1 and PC2 = No RPs within PC1 or within PC2 = Yes RPs between PC1/2 and PC3/4/5/6 = Yes RPs between PC3/4/5/6 = Yes					
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) Mem Read Transactions	No					
PCIe Controller (PC) Root Port (RP) Peer-2-Peer (P2P) MCTP VDM Transactions	RPs within PC1 or within PC2 = Yes RPs between PC1/2/3/4/5/6 = Yes					
PCIe Root Port Initiated Dynamic Width Change	No	No	No	No	No	No
PCIe Root Port Initiated Dynamic Speed Change	Yes	Yes	Yes	Yes	Yes	Yes
End Point Device Initiated Dynamic Width Change	Yes	Yes	Yes	Yes	Yes	Yes
End Point Device Initiated Dynamic Speed Change	Yes	Yes	Yes	Yes	Yes	Yes

NOTES

1. No restrictions on PCIe Controller. PCIe RAID is expected to work across all Root Ports within a PCIe Controller and between Root Ports from different PCIe Controllers.
2. No RAID support between PCIe and SATA storage devices.
3. The Intel® Rapid Storage Technology (RST) does not restrict RAID modes so if any unsupported RAID mode is enabled it is up to the motherboard designer to validate any non-supported RAID mode.

22.1.1 PCI Express* Power Management

S4/S5 Sleep State Support

Software initiates the transition to S4/S5 by performing an IO write to the Power Management Control register in the processor. After the IO write completion has been returned, 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 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 S4/ S5.

Latency Tolerance Reporting (LTR)

The PCIe Controller Root Ports support the extended Latency Tolerance Reporting (LTR) capability. LTR provides a means for device endpoints to dynamically report their service latency requirements for memory reads and write access's to the Root Ports through the Latency Tolerance Reporting messages. Endpoint devices should transmit a new LTR message to the Root Ports initially during boot and each time its latency tolerance changes. This latency information allows the Power Management Controller (PMC) to make effective and accurate decisions to transition the platform to deeper power management states without the cost of making the wrong decision, since deeper power management states are usually associated with longer exit latency.

22.1.2 Port 80h 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 80h cycles.

22.1.3 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.

22.1.4 Advanced Error Reporting

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

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

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

Alternative Routing - ID Interpretation (ARI)

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

Access Control Services (ACS)

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

22.1.6 PCI Express* Receiver Lane Polarity Inversion

The PCI Express* Base Specification requires polarity inversion to be supported independently by all receivers across a Link where 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 PCIe* Controllers to enable it.

NOTE

The 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".

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

22.2 Signal Description

Signal Name	Type	Description	Processor
PCIE_[20:1]_TXN **PCIE_[20:1]_TXP**	O	**PCI Express* Differential Transmit Pairs** These are the PCI Express* based outbound high-speed differential signals	**U**
PCIE_[20:1]_RXN **PCIE_[20:1]_RXP**	I	**PCI Express* Differential Receive Pairs** These are the PCI Express* based inbound high-speed differential signals	
PCIE_1_RCOMP **PCIE_2_RCOMP** **PCIE_3_RCOMP** **PCIE_5_RCOMP**	A	**PCI Express* PHY Impedance Compensation Inputs**	
PCIE_[28:1]_TXN **PCIE_[28:1]_TXP**	O	**PCI Express* Differential Transmit Pairs** These are the PCI Express* based outbound high-speed differential signals	**H**
PCIE_[28:1]_RXN **PCIE_[28:1]_RXP**	I	**PCI Express* Differential Receive Pairs** These are the PCI Express* based inbound high-speed differential signals	
PCIE_1_RCOMP **PCIE_2_RCOMP** **PCIE_3_RCOMP** **PCIE_5_RCOMP**	A	**PCI Express* PHY Impedance Compensation Inputs**	
GPP_H16/DDPB_CTRLCLK/ **PCIE_LINK_DOWN** /USB-C_GPP_H16	O	**PCI Express* Link Down Debug Signal** PCIe link failure debug signal. PCIe Root Port(s) will assert this signal when a link down event occurs and is detected. For example when a link fails to train during an L1 sub-state exit event.	**U/H**

22.3 I/O Signal Planes and States

Table 72. Power Plane and States for PCI Express* Signals

Signal Name	Type	Power Plane	During Reset[2]	Immediately After Reset[2]	S4/S5
PCIE_*_TXP/ PCIE_*_TXN	O	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
PCIE_*_RXP/ PCIE_*_RXN	I	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
PCIE_*_RCOMP	I	Primary	Undriven	Undriven	Undriven

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

22.4 PCI Express* Root Port Support Feature Details

Table 73. PCI Express* Root Port Feature Details

Processor	Max Transfer Rate	Max Devices (Root Ports)	Max Lanes	PCIe Gen Type	Encoding	Transfer Rate (MT/s)	Theoretical Max Bandwidth (GB/s)			
							x1	x2	x4	x8
U	16 GT/s (Gen4)	9 [2]	20	1	8b/10b	2500	0.25	0.50	1.00	N/A
				2	8b/10b	5000	0.50	1.00	2.00	N/A
				3	128b/130b	8000	1.00	2.00	3.94	N/A
				4	128b/130b	16000	1.97	3.94	7.88	N/A
H	32 GT/s (Gen5)	9 [2]	28	1	8b/10b	2500	0.25	0.50	1.00	2.46
				2	8b/10b	5000	0.50	1.00	2.00	4.92
				3	128b/130b	8000	1.00	2.00	3.94	7.88
				4	128b/130b	16000	1.97	3.94	7.88	15.75
				5	128b/130b	32000	3.94	7.88	15.75	31.51

Notes: 1. Theoretical Maximum Bandwidth (GB/s) = ((Transfer Rate * Encoding * # PCIe Lanes) /8)/1000
- Gen4 with 4 PCIe Lanes Example = ((16000 * 128/130* 4)/8)/1000 = 7.88 GB/s
- Gen5 with 8 PCIe Lanes Example = ((32000 * 128/130* 8)/8)/1000 = 31.51 GB/s

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

Figure 21. Processor-U Supported PCI Express* Link Configurations

Processor-U	SOC (System On Chip) Tile												IOE (IO Expander) Tile							
	FIA-2								FIA-3				FIA-4							
Flex I/O Lanes	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
PCIe Controller	1				2				3				4				5			
PCIe Max Rate	Gen4				Gen4				Gen4				Gen4				Gen4			
PCIe Lanes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PCIe Configurations (Bi-Furcation) [3]	1px4				1px4				1px4				1px4				1px4			
	1px4(LR)				1px4(LR)				1px4(LR)				1px4(LR)				1px4(LR)			
	2px2				2px2															
	2px2(LR)				2px2(LR)															
	1px2+2px1				1px2+2px1															
	2px1+1px2 [2]				2px1+1px2 [2]															
	4px1				4px1															
Logical Link Lanes	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
	0	1	0	1	0	1	0	1												
	1	0	1	0	1	0	1	0												
	0	1	0	0	0	1	0	0												
	0	0	1	0	0	0	1	0												
	0	0	0	0	0	0	0	0												
Assigned Root Ports	RP1				RP5				RP9				RP10				RP11			
	RP1				RP5				RP9				RP10				RP11			
	RP1		RP3		RP5		RP7													
	RP3		RP1		RP7		RP5													
	RP1		RP3	RP4	RP5		RP7	RP8												
	RP4	RP3	RP1		RP8	RP7	RP5													
	RP1	RP2	RP3	RP4	RP5	RP6	RP7	RP8												
Bus - Dev - Func (BDF) Assignments[1]	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func
	1	0h	1Ch	0h	5	0h	1Ch	4h	9	0h	6h	0h	10	0h	6h	1h	11	0h	6h	2h
	2	0h	1Ch	1h	6	0h	1Ch	5h												
	3	0h	1Ch	2h	7	0h	1Ch	6h												
	4	0h	1Ch	3h	8	0h	1Ch	7h												

Figure 22. Processor-H Supported PCI Express* Link Configurations

Processor-H	SOC (System On Chip) Tile												IOE (IO Expander) Tile															
	FIA-2								FIA-3				FIA-4								FIA-5							
Flex I/O Lanes	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
PCIe Controller	1				2				3				4				5				6							
PCIe Max Rate	Gen4				Gen4				Gen4				Gen4				Gen4				Gen5							
PCIe Lanes	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
PCIe Configurations (Bi-Furcation) [3]	1px4				1px4				1px4				1px4				1px4				1px8							
	1px4(LR)				1px4(LR)				1px4(LR)				1px4(LR)				1px4(LR)				1px8(LR)							
	2px2				2px2																							
	2px2(LR)				2px2(LR)																							
	1px2+2px1				1px2+2px1																							
	2px1+1px2 [2]				2px1+1px2 [2]																							
	4px1				4px1																							
Logical Link Lanes	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	7	6	5	4	3	2	1	0
	0	1	0	1	0	1	0	1																				
	1	0	1	0	1	0	1	0																				
	0	1	0	0	0	1	0	0																				
	0	0	1	0	0	0	1	0																				
	0	0	0	0	0	0	0	0																				
Assigned Root Ports	RP1				RP5				RP9				RP10				RP11				RP12							
	RP1				RP5				RP9				RP10				RP11				RP12							
	RP1		RP3		RP5		RP7																					
	RP3		RP1		RP7		RP5																					
	RP1		RP3	RP4	RP5		RP7	RP8																				
	RP4	RP3	RP1		RP8	RP7	RP5																					
	RP1	RP2	RP3	RP4	RP5	RP6	RP7	RP8																				
Bus - Dev - Func (BDF) Assignments[1]	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus	Dev	Func	RP	Bus		Dev		Func		
	1	0h	1Ch	0h	5	0h	1Ch	4h	9	0h	6h	0h	10	0h	6h	1h	11	0h	6h	2h	12	0h		1h		0h		
	2	0h	1Ch	1h	6	0h	1Ch	5h																				
	3	0h	1Ch	2h	7	0h	1Ch	6h																				
	4	0h	1Ch	3h	8	0h	1Ch	7h																				

NOTES

1. Device (BDF) groupings have multiple functions, the lowest active Root Port within the Device (BDF) grouping will always be assigned Function 0 while any remaining active Root Port within the Device (BDF) grouping will be assigned their mapped Function # as shown.
2. 2px1+1px2 is based off selecting 1px2+2px1 with Lane Reversal Enabled
3. Reduced Root Port width configurations, within Bi-Furcation configurations, are supported (example: x2 PCIe End Point Device populated in a PCIe Controller set as 1px4 will result in a 1px2 PCIe Root Port configuration or x1 PCIe End Point Device populated in a PCIe Controller set as 1px4 will result in a 1px1 PCIe Root Port configuration).
4. FIA = Flex-IO Adapter
5. The PCIe* Link Configuration support will vary depending on the SKU. Refer to the SKU details covered in the Introduction on page 17.
6. LR = Lane Reversal
7. PCIe Configuration (#p) x (#) = (Number of PCIe Root Ports) x (Number of Data Lane Pairs per PCIe Root Port)
8. RP# refers to a specific PCI Express* Root Port #; for example RP3 = PCI Express* Root Port 3
9. A PCIe* Lane is composed of a single pair of Transmit (TX) and Receive (RX) differential pairs. A connection between two PCIe* devices is known as a PCIe* Link, and is built up from a collection of one or more PCIe* Lanes which make up the width of the link (such as bundling 2 PCIe* Lanes together would make a x2 PCIe* Link). A PCIe* Link is addressed by the lowest number PCIe* Lane it connects to and is known as the PCIe* Root Port (such as a x2 PCIe* Link connected to PCIe* Lanes 3 and 4 would be called x2 PCIe* Root Port 3).
10. The PCIe* Lanes can be configured independently from one another but the max number of configured Root Ports (Devices) must not be exceeded
11. Unidentified lanes within a PCIe* Link Configuration are disabled but their physical lanes are used for the identified Root Port

23.0 Serial ATA (SATA)

The SATA controller support two modes of operation, AHCI mode using memory space and RAID mode. The SATA controller does not support IDE legacy mode using I/O space. Therefore, AHCI software is required. The SATA controller supports the Serial ATA Specification, Revision 3.2.

Not all functions and capabilities may be available on all SKUs. Refer to Introduction on page 17 for details on feature availability.

Table 74. 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

Table 75. 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

23.1 Functional Description

The SATA host controller supports AHCI or RAID mode.

The SATA controller does not support legacy IDE mode or combination mode.

The SATA controller interacts with an attached mass storage device through a register interface that is compatible with a SATA AHCI/RAID host adapter. The host software follows existing standards and conventions when accessing the register interface and follows standard command protocol conventions.

23.1.1 Features Supported

The 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 SATA controller register .

The SATA supports the following features:

- Port Multiplier:
  - The SATA controller may optionally support command-based switching Port Multipliers.

    NOTE

    BIOS must clear this bit if Port Multipliers are not supported.

- Host Initiated Loopback Mode:
  - The need to use PxCMD.CLO to clear the internal BSY bit before setting the PxCMD.ST to '1', as there is no power-up register FIS from device to clear the BSY bit.
  - Only one CI bit can be set per each Loopback session.
  - Only one PRD can be used in each CI.
  - Only PRD size of 256B is supported.
  - Only CL.CFL=0 is supported
  - Need to set the CL.P bit for the CI, to allow PRD prefetch, without waiting for device FIS.
  - Need to clear the PxCMD.ST bit after the loopback session done and set it again to start another session.

The SATA controller does **not** support:

- FIS Based Switching
- IDE mode or combination mode
- Cold Presence Detect

23.1.2 SATA 6 Gb/s Support

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

23.1.3 Hot Plug Operation

The 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 the power savings in the low power states while enabling Hot - Plug operation. Refer to Chapter 7 of the AHCI specification for details.

23.1.4 Intel® Rapid Storage Technology (Intel® RST)

The SATA controller provides support for Intel® Rapid Storage Technology, providing both AHCI and integrated RAID functionality. 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 Microsoft* Windows* compatible driver, and a user interface for configuration and management of the RAID capability of SATA controller.

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 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 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 Processor'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. Processor SKU enabled for Intel® Rapid Storage Technology.
2. Intel® Rapid Storage Technology RAID Option ROM or UEFI Driver must be on the platform.
3. Intel® Rapid Storage Technology drivers, most recent revision.
4. At least two SATA hard disk drives (minimum depends on RAID configuration).

Intel® Rapid Storage Technology is not available in the following configurations:

1. The SATA controller is programmed in RAID mode, but the AIE bit is set to 1.

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.

23.1.5 Power Management Operation

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

Power State Mappings

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

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

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

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

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

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

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

Power State Transitions

- **Partial and Slumber State Entry/Exit**

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

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

NOTE

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

- **Devslp State Entry/Exit**

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

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

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

- **Device D1 and D3 States**

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

- **Host Controller $D3_{HOT}$ State**

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

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

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

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

Low Power Platform Consideration

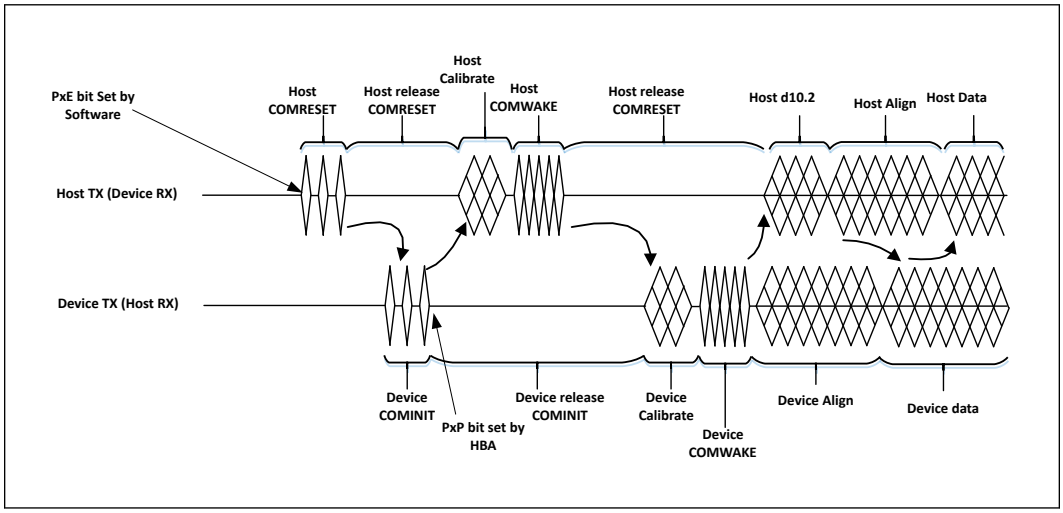
When low power feature is enabled, the Intel® SATA controller may power off PLLs or OOB detection circuitry while in the Slumber link power state. As a result, a device initiated wake may not be recognized by the host. For example, when the low power feature is enabled it can prevent a Zero Power ODD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

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

23.1.6 SATA Device Presence

The flow used to indicate SATA device presence is shown in the Figure below. The 'PxE' bit refers to bits, depending on the port being checked and the 'PxP' bits refer to the bits, depending on the port being checked. If the PCS/PxP bit is set a device is present, if the bit is cleared a device is not present. If a port is disabled, software can check to see 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 see if a new device is connected.

Figure 23. Port Enable/Device Present Bits Flow



23.1.7 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.

23.1.8 Advanced Host Controller Interface (AHCI) Operation

The 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 host—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 host/device designation for SATA devices—each device is treated as a host—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 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 to Section 7.3.1 of the AHCI Specification for more information.

23.2 Signal Description

Signal Name	Type	Description
GPP_E04/**SATA_DEVSLP0**/USB-C_GPP_E04	I or O	**Serial ATA Port [0] Device Sleep**: This is an open-drain pin on the side. Processor 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). Processor 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 0.
GPP_E05/**SATA_DEVSLP1**/ISH_GP7/USB-C_GPP_E05	I or O	**Serial ATA Port [1] Device Sleep**: This is an open-drain pin on the Processor side. Processor 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). Processor 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 1.
PCIE_1_TXN/**SATA_0_TXN** PCIE_1_TXP/**SATA_0_TXP**	O	**Serial ATA Differential Transmit Pair 0**: These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s.
PCIE_1_RXN/**SATA_0_RXN** PCIE_1_RXP/**SATA_0_RXP**	I	**Serial ATA Differential Receive Pair 0**: These inbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s.
PCIE_2_TXN/**SATA_1_TXN** PCIE_2_TXP/**SATA_1_TXP**	O	**Serial ATA Differential Transmit Pair 1** :These outbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s.
PCIE_2_RXN/**SATA_1_RXN** PCIE_2_RXP/**SATA_1_RXP**	I	**Serial ATA Differential Receive Pair 1**: These inbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s.

continued...

Signal Name	Type	Description
GPP_E00/SATAXPCIE0/**SATAGP0**/USB-C_GPP_E00	I or O	**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_E00.Pin defaults to Native mode as SATAXPCIE0 depends on soft-strap.
GPP_F10/SATAXPCIE1/**SATAGP1**/ISH_GP6A/USB-C_GPP_F10	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_F10.Pin defaults to Native mode as SATAXPCIE0 depends on soft-strap.
GPP_E08/DDPA_CTRLDATA/**SATALED#**/USB-C_GPP_E08	I or 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 VCC1P8 is required.

23.3 Integrated Pull-Ups and Pull-Downs

Signal Name	Resistor Type
SATAXPCIE[0:1]	Internal Pull-up
SATA_[0:1]_RXN	Internal Termination
SATA_[0:1]_RXP	Internal Termination
SATA_[0:1]_TXN	Internal Termination
SATA_[0:1]_TXP	Internal Termination
SATAGP[0:1]	Internal Termination
SATA_DEVSLP[0:1]	External Pull-up
Note: Internal Pull-Up Resistors are 20 kohm ± 30% unless specified.	

23.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[3]	Immediately after Reset[3]	S4/S5
SATA_[0:1]_TXN SATA_[0:1]_TXP SATA_[0:1]_RXN SATA_[0:1]_RXP	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down
SATALED#	Primary	Undriven	Undriven	Undriven
SATA_DEVSLP[0:1][1]	Primary	Undriven	Undriven	Undriven

continued...

Signal Name	Power Plane	During Reset[3]	Immediately after Reset[3]	S4/S5
SATAGP[0:1][2]	Primary	Undriven	Undriven	Undriven
SATAXPCIE[0:1][2]	Primary	Internal Pull-up	Internal Pull-up	Undriven

Notes: 1. Pin defaults to GPIO mode. The pin state during and immediately after reset follows default GPIO mode pin state. The pin state for S0 to S4/S5 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 S4/S5.

2. Pin defaults to Native mode as SATAXPCIEx depends on soft-strap.
3. Reset reference for primary well pins is RSMRST#.

24.0 Graphics

24.1 Processor Graphics

The processor graphics is based on Xe graphics core architecture that enables substantial gains in performance and lower-power consumption over prior generations. Xe architecture supports up to 8 Xe-core depending on the processor SKU.

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

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

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

24.1.1.1 Hardware Accelerated Video Decode

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

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

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

NOTES

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

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

Table 76. Hardware Accelerated Video Decoding

Codec	Profile	Level	Maximum Resolution
MPEG2	Main	Main - 15Mbps High - 40Mbps	FHD
AVC/H264	High Main Constrained Baseline	L5.2	4K
	4:2:0 8bit		4K @ 60
JPEG/MJPEG	Baseline	Unified level	16K x16K
HEVC/H265	Main12 420, 422, 444 - 8b/10b/12b SCC 420, 444 - 8b/10b	L6.1	8K @ 60(Decode Only) 8K@30(Decode Playback)
VP9	0 (420 8b) 1 (444 8b) 2 (420 10b/12b) 3 (444 10b/12b)	Unified level	8K @ 60(Decode only) 8K@30 (Decode Playback) 16Kx4K
AV1	Main (420 8-bit/10b)	L6.1	8K @ 60 (Video, Decode only) 8K@30 (Decode Playback) 16K x 16K (still picture)

Expected Performance: More than 16 simultaneous decode streams @ 1080p.

NOTE

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

24.1.1.2 Hardware Accelerated Video Encode

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

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

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

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

Table 77. Hardware Accelerated Video Encode

Codec	Profile	Level	Maximum Resolution
AVC/H264	High Main Constrained Baseline	L5.2	4K@60
JPEG			16Kx16K
HEVC/H265	Main10 422 - 8b/10b	L5.2	4K@60

continued...

Codec	Profile	Level	Maximum Resolution
	Main Main10 420, 444 - 8b/10b SCC 420 444 - 8b/10b	L6.1	4320p(8K) @60 16Kx12K
VP9	0 (420 8b) 1 (444 8b) 2 (420 10b) 3 (444 10b)	—	8K @30
AV1	Main (4:2:0 8b, 10b)	L6	8K@30

NOTE

Hardware encode for H264 SVC is not supported.

24.1.1.3 Hardware Accelerated Video Processing

There is hardware support for image processing functions such as De-interlacing, Film cadence detection, detail enhancement, gamut compression, Adaptive contrast enhancement, skin tone enhancement, total color control, De-noise, SFC (Scalar and Format Conversion), memory compression, Localized Adaptive Contrast Enhancement (LACE), 16 bpc support for de-noise/de-mosaic, Facial filter, HDR10 and Dolby Vision Tone Mapping HW acceleration.

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

- Direct3D* 11 Video API
- Intel® One VPL
- Intel® Graphics Control Library
- Intel VA API

NOTE

Not all features are supported by all the above APIs. Refer to the relevant documentation for more details.

24.1.1.4 Hardware Accelerated Transcoding

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

The processor graphics supports the following transcoding features:

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

24.1.2 Graphics Core Cache

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

First and Second Level Cache

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

Third Level Cache

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

24.2 Platform Graphics Hardware Feature

24.2.1 Hybrid Graphics

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

Microsoft* Hybrid Graphics definition includes the following:

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

25.0 Display

This chapter provides information on the following topics:

- Display Technologies Support
- Display Configuration
- Display Features

25.1 Display Technologies Support

Technology	Standard
eDP* 1.4b	VESA* Embedded DisplayPort* Standard 1.4b
DisplayPort* 2.1	VESA* DisplayPort* Standard 2.1
HDMI* 2.1	High-Definition Multimedia Interface Specification Version 2.1

Table 78. Display Ports Availability and Link Rate

Port	U/H-Series Processor
DDI A [4]	eDP* up to HBR3 DP* up to HBR3 HDMI* up to 6 Gbps
DDI B [4]	eDP* up to HBR3 DP* up to HBR3 HDMI* up to 6 Gbps
TCP 0	DP* up to UHBR20 HDMI* up to 12 Gbps
TCP 1	
TCP 2	
TCP 3	

Notes:
1. Dual Embedded panels supported on both Port A and B
2. MIPI DSI can be supported using on-board eDP to DSI bridge.
3. For non Type-C ports DisplayPort maximum supported link rate is HBR3.
4. DDI A is the primary port. In case, single eDP is used - only DDI A must be selected.
5. DDI eDP Port Configuration:
   a. Single eDP - Port A
   b. Dual eDP - Port A, Port B
      i. eDP Port B is only supported on dual eDP case.
      ii. For dual eDP case, port A needs to be the primary screen and port B to be the companion display explicitly.

25.2 Display Interfaces

This section provides information on the following topics:

- Digital Display Interface (DDI) Signals

- Digital Display Interface TCP Signals

25.2.1 Digital Display Interface DDI Signals

Table 79. Digital Display Interface DDI Signals

Signal Name	Type	Description
DDIA_TXP[3:0] DDIA_TXN[3:0]	O	Digital Display Interface A (DDIA): Digital Display Interface main link transmitter lanes.
DDIA_AUXP DDIA_AUXN	I/O	Digital Display Interface A (DDIA): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.
GPP_E08/**DDPA_CTRLDATA**/ SATALED#/USB-C_GPP_E08 GPP_E22/**DDPA_CTRLCLK**/ DNX_FORCE_RELOAD/USB-C_GPP_E22	I/O	Digital Display Interface A (DDIA): HDMI Graphics Management Bus (GMBUS).
GPP_E14/**DDSP_HPDA**/ DISP_MISCA/USB-C_GPP_E14	I	Digital Display Interface A (DDIA): Hot Plug Detect (HPD).
VDDEN	O	Digital Display Interface A (DDIA): eDP Panel power control enable signal.
BKLTEN	O	Digital Display Interface A (DDIA): eDP Panel back-light control enable signal.
BKLTCTL	O	Digital Display Interface A (DDIA): eDP Panel back-light control Pulse Wide Modulation (PWM) signal.
DDIB_TXP[3:0] DDIB_TXN[3:0]	O	Digital Display Interface B (DDIB): Digital Display Interface main link transmitter lanes.
DDIB_AUXP DDIB_AUXN	I/O	Digital Display Interface B (DDIB): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.
GPP_H17/**DDPB_CTRLDATA**/USB-C_GPP_H17 GPP_H16/**DDPB_CTRLCLK**/ PCIE_LINK_DOWN/USB-C_GPP_H16	I/O	Digital Display Interface B (DDIB): HDMI Graphics Management Bus (GMBUS).
GPP_B16/**DDSP_HPDB**/ DISP_MISCB/USB-C_GPP_B16	I	Digital Display Interface B (DDIB): Hot Plug Detect (HPD).
GPP_B17/**VDDEN2**/USB-C_GPP_B17	O	Digital Display Interface B (DDIB): eDP Panel power control enable signal.
GPP_D01/I2C3A_SDA/**BKLTEN2**/ ISH_I2C2A_SDA/USB-C_GPP_D01	O	Digital Display Interface B (DDIB): eDP Panel back-light control enable signal.
GPP_D02/I2C3A_SCL/**BKLTCTL2**/ ISH_I2C2A_SCL/USB-C_GPP_D02	O	Digital Display Interface B (DDIB): eDP Panel back-light control Pulse Wide Modulation (PWM) signal.
DDI_RCOMP	Analog	DDI IO Compensation resistors.
Notes: • Auxiliary Channel (AUX CH) is a half-duplex bidirectional channel used for link management and device control. AUX CH is an AC coupled differential signal. • GMBUS follows I2C Protocol.		

25.2.2 Digital Display Interface TCP Signals

Table 80. Digital Display Interface TCP Signals

Signal Name	Type	Description	Availability
TCP0_TXRX[1:0]_P TCP0_TXRX[1:0]_N TCP0_TX[1:0]_P TCP0_TX[1:0]_N	O	Digital Display Interface 0 (TCP0): Digital Display Interface main link transmitter lanes.	H/U-Series Processors
TCP0_AUX_P TCP0_AUX_N	I/O	Digital Display Interface 0 (TCP0): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.	H/U-Series Processors
GPP_C17/ TBT_LSX0_RXD/ **DDP0_CTRLDATA**/ USB-C_GPP_C17 GPP_C16/ TBT_LSX0_TXD/ **DDP0_CTRLCLK**/USB-C_GPP_C16	I/O	Digital Display Interface 0 (TCP0): HDMI Graphics Management Bus (GMBUS).	H/U-Series Processors
GPP_B09/ **DDSP_HPD0**/ DISP_MISC1/USB-C_GPP_B09	I	Digital Display Interface 0 (TCP0): Hot Plug Detect (HPD).	H/U-Series Processors
TCP1_TXRX[1:0]_P TCP1_TXRX[1:0]_N TCP1_TX[1:0]_P TCP1_TX[1:0]_N	O	Digital Display Interface 1 (TCP1): Digital Display Interface main link transmitter lanes.	H/U-Series Processors
TCP1_AUX_P TCP1_AUX_N	I/O	Digital Display Interface 1 (TCP1): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.	H/U-Series Processors
GPP_C19/ TBT_LSX1_RXD/ **DDP1_CTRLDATA**/ USB-C_GPP_C19 GPP_C18/ TBT_LSX1_TXD/ **DDP1_CTRLCLK**/USB-C_GPP_C18	I/O	Digital Display Interface 1 (TCP1): HDMI Graphics Management Bus (GMBUS).	H/U-Series Processors
GPP_B10/ **DDSP_HPD1**/ DISP_MISC2/USB-C_GPP_B10	I	Digital Display Interface 1 (TCP1): Hot Plug Detect (HPD).	H/U-Series Processors
TCP2_TXRX[1:0]_P TCP2_TXRX[1:0]_N TCP2_TX[1:0]_P TCP2_TX[1:0]_N	O	Digital Display Interface 2 (TCP2): Digital Display Interface main link transmitter lanes.	H/U-Series Processors only
TCP2_AUX_P TCP2_AUX_N	I/O	Digital Display Interface 2 (TCP2): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.	H/U-Series Processors only

continued...

Signal Name	Type	Description	Availability
GPP_C21/ TBT_LSX2_RXD/ **DDP2_CTRLDATA**/ USB-C_GPP_C21 GPP_C20/ TBT_LSX2_TXD/ **DDP2_CTRLCLK**/USB-C_GPP_C20	I/O	Digital Display Interface 2 (TCP2): HDMI Graphics Management Bus (GMBUS).	H/U-Series Processors only
GPP_B11/USB_OC1#/ **DDSP_HPD2**/ DISP_MISC3/USB-C_GPP_B11	I	Digital Display Interface 2 (TCP2): Hot Plug Detect (HPD).	H/U-Series Processors only
TCP3_TXRX[1:0]_P TCP3_TXRX[1:0]_N TCP3_TX[1:0]_P TCP3_TX[1:0]_N	O	Digital Display Interface 3 (TCP3): Digital Display Interface main link transmitter lanes.	H/U-Series Processors only
TCP3_AUX_P TCP3_AUX_N	I/O	Digital Display Interface 3 (TCP3): DisplayPort Auxiliary: Half-duplex, bidirectional channel consist of one differential pair.	H/U-Series Processors only
GPP_C23/ TBT_LSX3_RXD/ **DDP3_CTRLDATA**/ USB-C_GPP_C23 GPP_C22/ TBT_LSX3_TXD/ **DDP3_CTRLCLK**/USB-C_GPP_C22	I/O	Digital Display Interface 3 (TCP3): HDMI Graphics Management Bus (GMBUS).	H/U-Series Processors only
GPP_B14/USB_OC2#/ **DDSP_HPD3**/ DISP_MISC4/USB-C_GPP_B14	I	Digital Display Interface 3 (TCP3): Hot Plug Detect (HPD).	H/U-Series Processors only
TCP_RCOMP	Analog	DDI IO Compensation resistors.	H/U-Series Processors
Notes: • Auxiliary Channel (AUX CH) is a half-duplex bidirectional channel used for link management and device control. AUX CH is an AC coupled differential signal. • GMBUS follows I2C Protocol.			

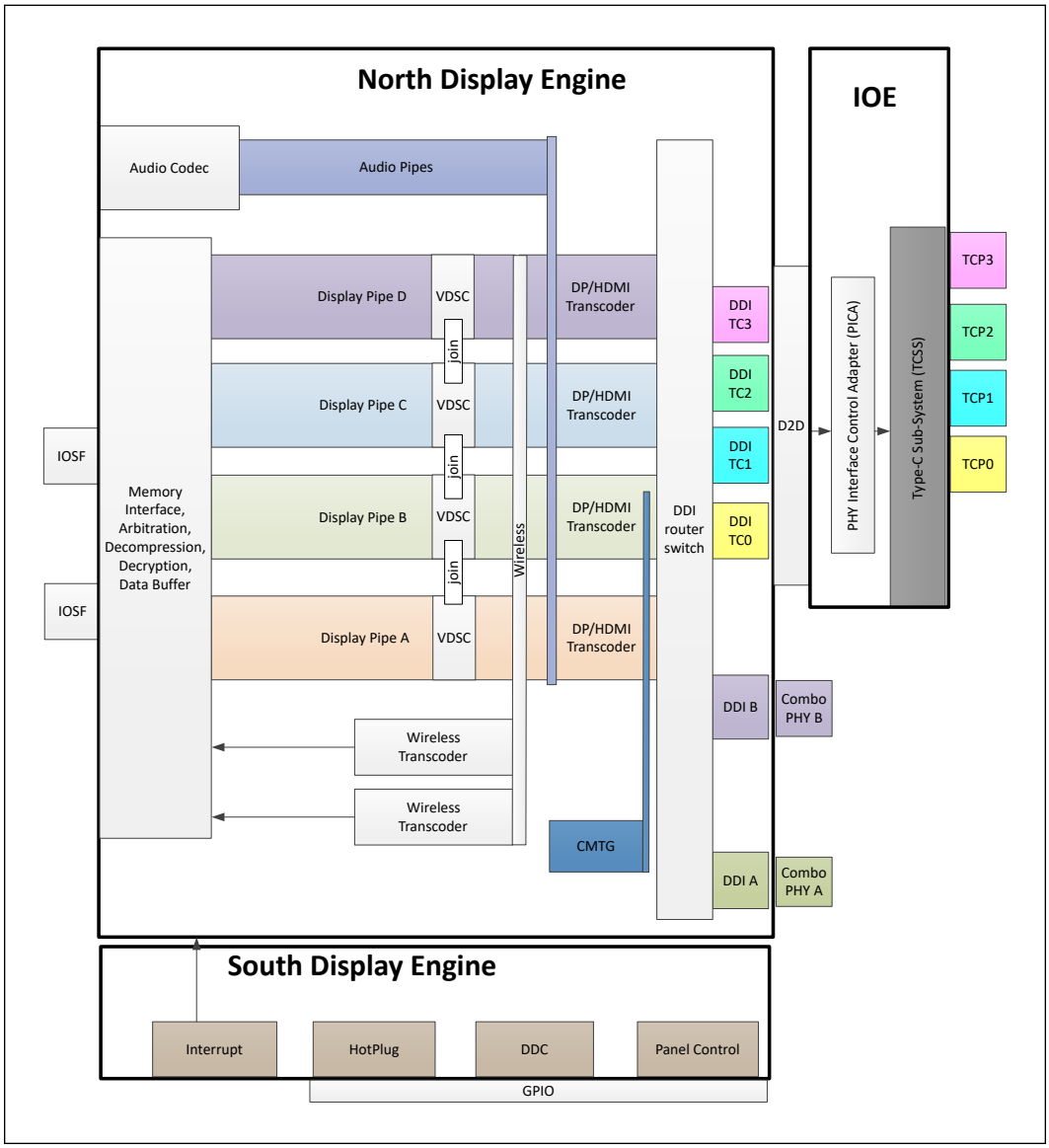
25.3 Display Features

This section provides information on the following topics:

- General Capabilities
- Multiple Display Configurations
- High-bandwidth Digital Content Protection (HDCP)
- DisplayPort*
- High-Definition Multimedia Interface (HDMI*)
- embedded DisplayPort* (eDP*)
- Integrated Audio

25.3.1 General Capabilities

Figure 24. Processor Display Architecture



NOTE

For port availability in each of the processor series, refer to Table 78 on page 204.

- Up to four simultaneous displays, 4K60Hz Embedded panel concurrent with:
  - Single external panel up to 8K60Hz, supported by joining two pipes over single port.
  - Up to 3x4K60Hz External panel.
- Display interfaces supported:

  - DDI interfaces supports DP*, HDMI*, eDP*
  - TCP interfaces supports DP*, HDMI*, Display Alt Mode over Type-C and DP* tunneled.
- End-To-End (E2E) compression, Unified memory compression across GT, media and display.
- Audio stream support on external ports.
- HDR (High Dynamic Range) support.
- Four Display Pipes - Supporting blending, color adjustments, scaling and dithering.
- Transcoder - Containing the Timing generators supporting eDP*, DP*, HDMI* interfaces.
- One Low Power optimized pipes supporting Embedded DisplayPort*
  - LACE (Localized Adaptive Contrast Enhancement), supported up to 5 K resolutions.
  - 3D LUT - power efficient pixel modification function for color processing.
  - FBC (Frame Buffer Compression) - power saving feature.

25.3.2 Multiple Display Configurations

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

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

25.3.3 High-bandwidth Digital Content Protection (HDCP)

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

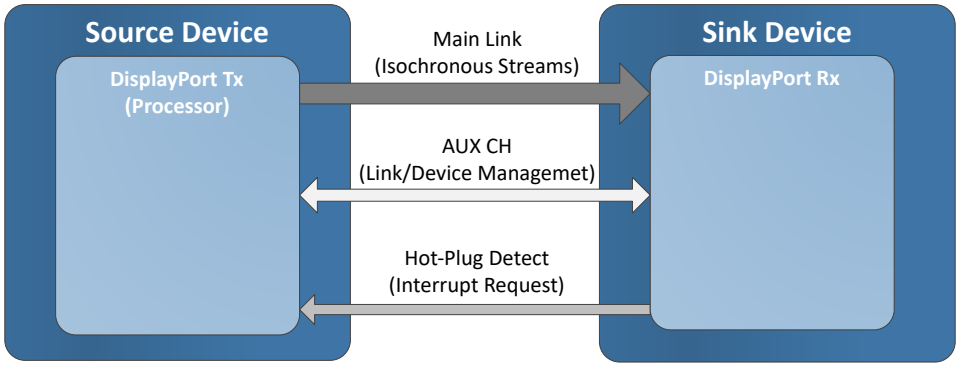
25.3.4 DisplayPort*

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

A DisplayPort* consists of a Main Link (four lanes), Auxiliary channel, and a Hot-Plug Detect signal. The Main Link is a unidirectional, high-bandwidth, and low-latency channel used for transport of isochronous data streams such as uncompressed video and audio. The Auxiliary Channel (AUX CH) is a half-duplex bi-directional channel used for link management and device control. The Hot-Plug Detect (HPD) signal serves as an interrupt request from the sink device to the source device.

The processor is designed in accordance with VESA* DisplayPort* specification.

Figure 25. DisplayPort* Overview



- Supports main link of 1, 2, or 4 data lanes.
- Link rate supports up to UHBR20 (UHBR13.5 is not supported)
- Aux channel for Link/Device management
- Hot Plug Detect
- Supports up to 36 BPP (Bit Per Pixel)
- Supports SSC
- Supports YCbCR 4:4:4, YCbCR 4:2:0, YCbCR 4:2:2, and RGB color format
- Supports MST (Multi-Stream Transport)
- Supports VESA DSC 1.2b
- Supports panel replay
- Adaptive Sync

25.3.4.1 Multi-Stream Transport (MST)

- The processor supports Multi-Stream Transport (MST), enabling multiple monitors to be used via a single DisplayPort connector.
- Maximum MST DP supported resolution:

Table 81. Display Resolutions and Link Bandwidth for Multi-Stream Transport Calculations

Pixels per Line	Lines	Refresh Rate [Hz]	Pixel Clock [MHz]	Link Bandwidth [Gbps]
1920	1080	60	148.5	4.46
1920	1200	60	154	4.62
2048	1152	60	156.75	4.70
2048	1280	60	174.25	5.23
2048	1536	60	209.25	6.28

continued...

Pixels per Line	Lines	Refresh Rate [Hz]	Pixel Clock [MHz]	Link Bandwidth [Gbps]
2304	1440	60	218.75	6.56
2560	1440	60	241.5	7.25
3840	2160	30	262.75	7.88
2560	1600	60	268.5	8.06
2880	1800	60	337.5	10.13
3200	2400	60	497.75	14.93
3840	2160	60	533.25	16.00
4096	2160	60	556.75	16.70
4096	2304	60	605	18.15
5120	3200	60	1042.5	31.28

Notes: 1. All the above is related to bit depth of 24.
2. The data rate for a given video mode can be calculated as
   Data Rate = Pixel Frequency * Bit Depth
3. The bandwidth requirements for a given video mode can be calculated as: Bandwidth = Data Rate * 1.25 (for 8b/10b coding overhead).
4. The link bandwidth depends if the standards is reduced blanking or not.
   If the standard is not reduced blanking - the expected bandwidth may be higher.
   For more details, refer to VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMT). Version 1.0, Rev. 13 February 8, 2013
5. To calculate what are the resolutions that can be supported in MST configurations, follow the below guidelines:
   a. Identify what is the link bandwidth column according to the requested display resolution.
   b. Summarize the bandwidth for two of three displays accordingly, and make sure the final result is below 21.6 Gbps. (for example: 4 lanes HBR2 bit rate)
   For example:
   a. Docking two displays: 3840x2160@60 Hz + 1920x1200@60 Hz = 16 + 4.62 = 20.62 Gbps [Supported]
   b. Docking three displays: 3840x2160@30 Hz + 3840x2160@30 Hz + 1920x1080@60 Hz = 7.88 + 7.88 + 4.16 = 19.92 Gbps [Supported].
6. MST bandwidth number is calculated without VESA Display Stream Compression (VDSC).

Table 82. DisplayPort Maximum Resolution

Standard	U/H-Series Processor
DP*	8K60Hz compressed, 5K120Hz compressed

Notes: 1. bpp - bit per pixel.
2. Resolution support is subject to memory BW availability.
3. High resolutions will consume two display pipes.
4. For non Type-C ports, DisplayPort maximum supported link rate is HBR3.

25.3.5 High-Definition Multimedia Interface (HDMI*)

The High-Definition Multimedia Interface (HDMI*) is provided for transmitting digital audio and video signals from DVD players, set-top boxes, and other audio-visual sources to television sets, projectors, and other video displays. It can carry high-quality multi-channel audio data and all standard and high-definition consumer electronics video formats. The HDMI display interface connecting the processor and display devices uses transition minimized differential signaling (TMDS) or Fixed Rate Link (FRL) to carry audiovisual information through the same HDMI cable.

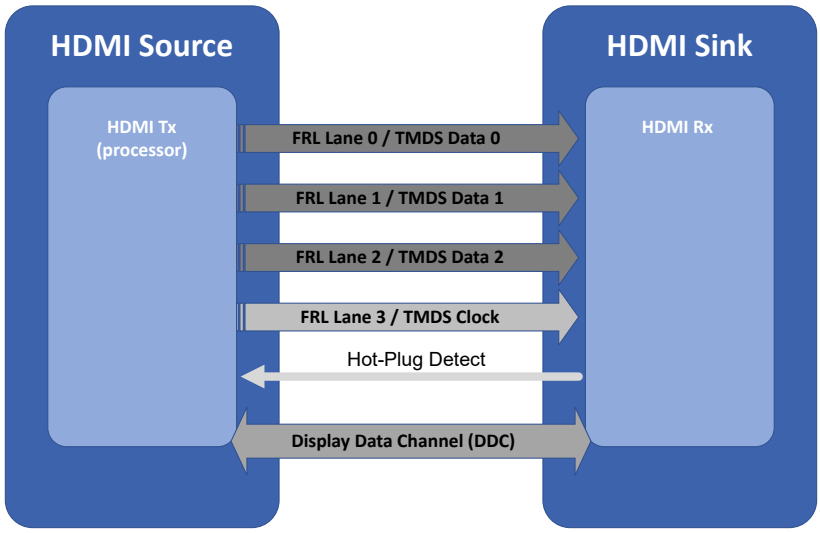
HDMI* includes three separate communications channels: TMDS or FRL, DDC/GMBUS, and the optional CEC (consumer electronics control). CEC is not supported on the processor. As shown in the following figure, the HDMI* cable carries four differential pairs that make up the TMDS data and clock channels or FRL lanes. These channels are used to carry video, audio, and auxiliary data. In addition, HDMI carries a VESA DDC. The DDC/GMBUS is used by an HDMI* Source to determine the capabilities and characteristics of the Sink.

Audio, video, and auxiliary (control/status) data is transmitted across the three TMDS data channels.

TDMS architecture has 3 Data lane and 1 Clock lane

FRL architecture has 4 Data lane, and no clock lane

Figure 26. HDMI* Overview



- Supports up to 6Gbps TMDS link rates on 3 lanes
- Supports up to 12Gbps FRL link rates on 4 lanes
- Support YCbCR 4:4:4, YCbCR 4:2:0, YCbCR 4:2:2, and RGB color format
- Supports up to 36 BPP (Bit Per Pixel)
- Supports VESA DSC 1.2a in FRL mode
- Hot Plug Detect
- Adaptive Sync supported in FRL mode

Table 83. HDMI Maximum Resolution

Standard	U/H-Series Processor
HDMI 2.1 TMDS (Up to 6Gbps)	4K60
HDMI 2.1 FRL (Up to 12 Gbps)	4K120 8K60Hz compressed,

continued...

Standard	U/H-Series Processor
	5K120Hz compressed
Notes: 1. bpp - bit per pixel. 2. Resolution support is subject to memory BW availability. 3. Compressed mean DSC only	

25.3.6 embedded DisplayPort* (eDP*)

The embedded DisplayPort* (eDP*) is an embedded version of the DisplayPort standard oriented towards applications such as notebook and All-In-One PCs. Like DisplayPort, embedded DisplayPort* also consists of the Main Link, Auxiliary channel, and an optional Hot-Plug Detect signal.

- Supports Low power optimized pipes
- Supports up to HBR3 link rate
- Supports Backlight PWM control and enable signals, and power enable
- Supports VESA DSC 1.2a
- Supports SSC
- Panel Self Refresh 1
- Panel Self Refresh 2
- MSO 2x2, 4x1(Multi Segment Operation)
- Dedicated Aux channel
- Adaptive Sync

Table 84. Embedded DisplayPort Maximum Resolution

Standard	U/H-Series Processor
eDP*	4K120Hz HDR
Notes: 1. Maximum resolution is based on the implementation of 4 lanes at HBR3 link data rate. 2. Resolution support is subject to memory BW availability.	

25.3.7 Integrated Audio

- HDMI* and DisplayPort interfaces can carry audio along with video.
- The processor supports up to four High Definition Audio streams on four digital ports simultaneously.

Table 85. Processor Supported Audio Formats over HDMI* and DisplayPort*

Audio Formats	HDMI*	DisplayPort*
AC-3 Dolby* Digital	Yes	Yes
Dolby* Digital Plus	Yes	Yes
DTS-HD*	Yes	Yes
LPCM, 32 KHz, 44.1 KHz, 48 KHz, 88.2 KHz, 96 KHz, 176.4 KHz, and 192 KHz, 16/24 bit, 2/4/6/8 channels	Yes	Yes
Dolby* TrueHD, DTS-HD Master Audio* (Lossless Blu-Ray Disc* Audio Format)	Yes	Yes

The processor will continue to support Silent stream. A Silent stream is an integrated audio feature that enables short audio streams, such as system events to be heard over the HDMI* and DisplayPort* monitors. The processor supports silent streams over the HDMI and DisplayPort interfaces at 32 KHz, 44.1 KHz, 48 KHz, 88.2 KHz, 96 KHz, 176.4 KHz, and 192 KHz sampling rates and silent multi-stream support.

26.0 Processor Sideband Signals

The sideband signals are used for the communication between the interfaces within the processor.

Table 86. Acronyms

Acronyms	Description
PECI	Platform Environmental Control Interface

26.1 Signal Description

Signal Name	Type	Description
THERMTRIP#	O	Signal from the processor to indicate that a thermal overheating has occurred.
PECI	I/O	Single-wire serial bus for accessing processor digital thermometer
GPP_E03/**PROC_GP0**/USB-C_GPP_E03	I	Thermal management signal
GPP_D03/**PROC_GP1**/USB-C_GPP_D03	I	Thermal management signal
GPP_E15/**PROC_GP2**/RSVD/ ISH_GP5A/USB-C_GPP_E15	I	Thermal management signal
GPP_E16/**PROC_GP3**/VRALERT#/ ISH_GP10/USB-C_GPP_E16	I	Thermal management signal

NOTE

If THERMTRIP# goes active, the processor is indicating an overheat condition. PROC_GP can be used from external sensors for the thermal management.

26.2 Integrated Pull-Ups and Pull-Downs

None

26.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset [1]	S4/S5
THERMTRIP#	Primary	Undriven	Undriven	OFF
PECI	Primary	Undriven	Undriven	OFF
PROC_GP[3:0]	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

27.0 General Purpose Input and Output

The General Purpose Input/Output (GPIO) signals are grouped into multiple groups (such as GPP_A, GPP_B, and so on). All GPIO groups are powered by the Primary well.

The high level features of GPIO:

- 1.8 V operation (including the muxed functions on the pin).
- Integrated pull-up / pull-down.
- Configurable as GPIO input, GPIO output, or native function signal.
- Configurable GPIO pad ownership by host, CSME, or ISH.
- SCI (GPE) and IOAPIC interrupt capable on all GPIOs
- NMI and SMI capability capable (on selected GPIOs).
- PWM, Serial Blink capable (on selected GPIOs).

Table 87. Acronyms

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

27.1 Functional Description

27.1.1 Interrupt / IRQ via GPIO Requirement

A GPIO, as an input, can be used to generate an interrupt / IRQ to the processor. In this case, it is required that the pulse width on the GPIO must be at least 100 us for the processor to recognize the interrupt.

27.1.2 Integrated Pull-ups and Pull-downs

All GPIOs have programmable internal pull-up/pull-down resistors. Most of the GPIOs have the internal pull-up/pull-down being off by default. The internal pull-up/pull-down for each GPIO can be enabled by BIOS programming the corresponding PAD_CFG_DW1 register, TERM bit.

27.1.3 SCI / SMI# and NMI

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

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

- GPP_B14, GPP_B20, GPP_B23

- GPP_C[23:22]
- GPP_D[04:00]
- GPP_E[08:00] ; GPP_E[16:13]

27.1.4 Timed GPIO

The processor supports two Timed GPIOs as native function (TIME_SYNC) that is multiplexed on GPIO pins. The intent usage of the Timed GPIO function is for time synchronization purpose.

Timed GPIO can be an input or an output:

- As an input, a GPIO input event triggers the HW to capture the processor Always Running Timer (ART) time in the Time Capture register. The GPIO input event must be asserted for at least two crystal oscillator clocks period in order for the event to be recognized.
- As an output, a match between the ART time and the software programmed time value triggers the HW to generate a GPIO output event and capture the ART time in the Time Capture register. If periodic mode is enabled, HW generates the periodic GPIO events based on the programmed interval. The GPIO output event is asserted by HW for at least two crystal oscillator clock periods.

NOTE

TIME_SYNC can be set as input when both Direction (DIR) bit and Enable (EN) bit in Timed GPIO Control Register are set to 1. When EN bit is set to 0, TIME_SYNC will default to output low regardless of DIR bit setting.

Timed GPIO supports event counter. When Timed GPIO is configured as input, event counter increments by one for every input event triggered. When Timed GPIO is configured as output, event counter increments by one for every output event generated. The event counter provides the correlation to associate the Timed GPIO event (the nth event) with the captured ART time. The event counter value is captured when a read to the Time Capture Value register occurs.

NOTE

When Timed GPIO is enabled, the crystal oscillator will not be shut down as crystal clock is needed for the Timed GPIO operation. As a result, SLP_S0# will not be asserted. This has implication to platform power (such as IDLE or S0ix power). Software should only enable Timed GPIO when needed and disable it when Timed GPIO functionality is not required.

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

Certain GPIOs are capable of supporting blink (BK, or also referred to as PWM), and serial blink (SBK, or also referred to as Serial POST Codes). The BK and SBK are implemented as native functions muxed on selected GPIOs. To enable BK or SBK on a GPIO having the capability, BIOS needs to select the BK or SBK native function on the GPIO.

BK provides a programmable PWM wave. The Blink/PWM frequency and duty cycle is programmable through the PWM Control register.

SBK allows system software to serialize POST or other messages on GPIO to a serial monitor. The Serial Blink messages is programmed through the Serial Blink Command/ Status and Serial Blink Data registers.

27.1.6 GPIO Ownership

A GPIO can be owned by the host, the Intel® CSME, or ISH depending on how the pin ownership being programmed. The programmed agent will then own the pin exclusively. For example, when a GPIO pad ownership is programmed to Intel® CSME or ISH, the host software no longer has access to the pin programming.

27.1.7 Native Function and TERM Bit Setting

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

Table 88. Native Function Signals Supporting Dynamic Termination Override

Native Function	Signal With Dynamic Termination Override
Intel®HD Audio	HDA_SDI[0:1], HDA_SDO, HDA_SYNC, I2S[5:0]_SCLK, I2S[2:0]_SFRM, I2S[2:0]_RXD, DMIC_DATA[1:0], SNDW[3:0]_DATA
Power Management	ACPRESENT, WAKE#, SOC_WAKE#
Touch Host Controller (THC)	THC0_SPI1_IO[3:0], THC0_SPI2_IO[3:0] THC1_SPI2_IO[3:0]
Thunderbolt / BSSB	TBT_LSX[3:0], BSSB_LS0_RX, BSSB_LS0_TX
I^3C	I3C[1:0]_SDA, I3C[1:0]_SCL, I3C1A_SDA, I3C1A_SCL
ISH	ISH_I3C0_SDA, ISH_I3C0_SCL

27.2 Signal Description

GPP_D22 and GPP_D23 have additional native function, BPKI3C_SDA (inout) and BPKI3C_SCL(inout), multiplexed onto the pins respectively as Native Function 1. The additional muxing is to support I3C debug capability which is new to H-Processor Series. The H-Processor GPP_D22 and GPP_D23 and their muxed functions are shown below:

GPP_D22/BPKI3C_SDA/BSSB_LS0_RX/USB-C_GPP_D22

GPP_D23/BPKI3C_SCL/BSSB_LS0_TX/USB-C_GPP_D23

The pins default to Native Function 1 or Native Function 2 depending on the HW pin strap GPP_F21/USB-C_GPP_F21.

For GPIO pin implementation including multiplexed native functions, default values, signal states, and other characteristics, download the pdf, click on the navigation pane and refer the spreadsheet, **842704-002_U_H_GPIO.xlsx**.

28.0 Interrupt Timer Subsystem (ITSS)

Table 89. Acronyms

Acronym	Description
ITSS	Interrupt Timer Subsystem
HPET	High Precision Event Timer
8254 PIT	Legacy 8254 Programmable Interrupt Timer
INTR	Interrupt
NMI	Non-maskable Interrupt
INIT	Processor Initialization
SERR	System Error

Table 90. References

Specification	Document Number/Location
ACPI Specification, Rev 5.0a	https://uefi.org/acpi/specs

28.1 Feature Overview

ITSS supports following features:

- It houses the HPET, Legacy 8254 Timers and APIC Interrupt Controllers.
- Fully synchronous-based design adopted for 8254 PIT.
- Functions as a simple Internal Host Space Error Collector and Reporting Block.
- 8254 PIT - Consists of 3 16-bit Timers capable of supporting up to 6 different modes.
- APIC - Supports up to 120 IRQs.
- HPET - Contains 8 Timer Blocks and a single always running 64-bit counter. Each Timer is interrupt capable, with option to route to APIC or directly to hose using MSI. Improved resolution, reduced overhead in comparison to Legacy 8254, IOxAPIC & RTC Timers.

28.2 Functional Description

The ITSS (Interrupt Timer Sub System) have below sub blocks:

- **ITSS** : Consists of the HPET, 8254 and APIC.

28.2.1 8254 Timers

There are three counters that have fixed uses. All registers and functions associated with these timers are in the Primary well. The 8254 unit is clocked by a 1.193 MHz periodic timer tick, which is functional only in S0 states. The 1.193 MHz periodic timer tick is generated off the XTAL clock.

Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is typically programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value 1 counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two each counter period. The counter negates IRQ0 when the count value reaches 0. It then reloads the initial count value and again decrements the initial count value by two each counter period. The counter then asserts IRQ0 when the count value reaches 0, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

28.2.1.1 Timer Programming

The counter/timers are programmed in the following fashion:

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

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

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

If a counter is programmed to read/write 2-byte counts, the following precaution applies – a program must not transfer control between writing the first and second Byte to another routine, which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of counter. Several commands are available:

- **Control Word Command:** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command:** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command:** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

The table below lists the six operating modes for the interval counters:

Table 91. 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.

28.2.1.2 Reading from Interval Timer

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

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for 2-byte counts, 2-bytes must be read. The2-bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

Simple Read

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

NOTE

Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations.

Counter Latch Command

The Counter Latch command, written to Port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a 2-byte count. The count value is then read from each counter's Count register as was programmed by the Control register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

If a Counter is latched and then, sometime later, latched again before the count is read, the second Counter Latch command is ignored. The count read is the count at the time the first Counter Latch command was issued.

Read Back Command

The Read Back command, written to Port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both the count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. If multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both the count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, returns the latched count. Subsequent reads return unlatched count.

28.2.2 APIC Advanced Programmable Interrupt Controller

The APIC is accessed via an indirect addressing scheme. These registers are mapped into memory space. These are programmable through PCI Config IOAC register.

28.2.3 High Precision Event Timer (HPET)

This function provides a set of timers that can be used by the operating system. The timers are defined such that the operating system may assign specific timers to be used directly by specific applications. Each timer can be configured to cause a separate interrupt.

The processor provides eight timers. The timers are implemented as a single counter with a set of comparators. Each timer has its own comparator and value register. The counter increases monotonically. Each individual timer can generate an interrupt when the value in its value register matches the value in the main counter.

Timer 0 supports periodic interrupts.

The registers associated with these timers are mapped to a range in memory space (much like the I/O APIC). However, it is not implemented as a standard PCI function. The BIOS reports to the operating system the location of the register space using

ACPI. The hardware can support an assignable decode space; however, BIOS sets this space prior to handing it over to the operating system. It is not expected that the operating system will move the location of these timers once it is set by BIOS.

28.2.3.1 Timer Accuracy

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

Within any 100 us period, the timer reports a time that is up to two ticks too early or too late. Each tick is less than or equal to 100 ns; thus, this represents an error of less than 0.2%.

The timer is monotonic. It does not return the same value on two consecutive reads (unless the counter has rolled over and reached the same value).

The main counter uses the XTAL as its clock. The accuracy of the main counter is as accurate as the crystal that is used in the system.

28.2.3.2 Timer Off-load

The timer off-load feature allows the HPET timers to remain operational during very low power S0 operational modes when the XTAL clock is disabled. The clock source during this off-load is the Real Time Clock's 32.768 kHz clock. This clock is calibrated against the XTAL clock during boot time to an accuracy that ensures the error introduced by this off-load is less than 10 ppb (0.000001%).

When the XTAL clock is active, the 64 bit counter will increment by one each cycle of the XTAL clock when enabled. When the XTAL clock is disabled, the timer is maintained using the RTC clock. The long-term (> 1 ms) frequency drift allowed by the HPET specification is 500 ppm. The off-load mechanism ensures that it contributes < 1 ppm to this, which will allow this specification to be easily met given the clock crystal accuracies required for other reasons.

Timer off-load is prevented when there are HPET comparators active.

The HPET timer runs typically on the XTAL crystal clock and is off-loaded to the 32 kHz clock once the processor enters C10. This is the state where there are no C10 wake events pending and when the off-load calibrator is not running. HPET timer re-uses this 28 bit calibration value calculated by PMC when counting on the 32 kHz clock. During C10 entry, PMC sends an indication to HPET to off-load and keeps the indication active as long as the processor is in C10 on the 32 kHz clock. The HPET counter will be off-loaded to the 32 kHz clock domain to allow the XTAL clock to shut down when it has no active comparators.

Theory of Operation

The Off-loadable Timer Block consists of a 64 bit fast clock counter and an 82 bit slow clock counter. During fast clock mode the counter increments by one on every rising edge of the fast clock. During slow clock mode, the 82 bit slow clock counter will increment by the value provided by the Off-load Calibrator.

The Off-loadable Timer will accept an input to tell it when to switch to the slow RTC clock mode and provide an indication of when it is using the slow clock mode. The switch will only take place on the slow clock rising edge, so for the 32 kHz RTC clock the maximum delay is around 30 us to switch to or from slow clock mode. Both of these flags will be in the fast clock domain.

When transitioning from fast clock to slow clock, the fast clock value will be loaded into the upper 64 bits of the 82 bit counter, with the 18 LSBs set to zero. The actual transition though happens in two stages to avoid metastability. There is a fast clock sampling of the slow clock through a double flop synchronizer. Following a request to transition to the slow clock, the edge of the slow clock is detected and this causes the fast clock value to park. At this point the fast clock can be gated. On the next rising edge of the slow clock, the parked fast clock value (in the upper 64 bits of an 82 bit value) is added to the value from the Off-load Calibrator. On subsequent edges while in slow clock mode the slow clock counter increments its count by the value from the Off-load Calibrator.

When transitioning from slow clock to fast clock, the fast clock waits until it samples a rising edge of the slow clock through its synchronizer and then loads the upper 64 bits of the slow clock value as the fast count value. It then de-asserts the indication that slow clock mode is active. The 32 kHz clock counter no longer counts. The 64 bit MSB will be over-written when the 32 kHz counter is reloaded once conditions are met to enable the 32 kHz HPET counter but the 18 bit LSB is retained and it is not cleared out during the next reload cycle to avoid losing the fractional part of the counter.

After initiating a transition from fast clock to slow clock and parking the fast counter value, the fast counter no longer tracks. This means if a transition back to fast clock is requested before the entry into off-load slow clock mode completes, the Off-loadable Timer must wait until the next slow clock edge to restart. This case effectively performs the fast clock to slow clock and back to fast clock on the same slow clock edge.

28.2.3.3 Periodic Versus Non-Periodic Modes

Non-Periodic Mode

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

When a timer is set up for non-periodic mode, it will generate an interrupt when the value in the main counter matches the value in the timer’s comparator register. Another interrupt will be generated when the main counter matches the value in the timer’s comparator register after a wrap around.

During run-time, the value in the timer’s comparator value register will not be changed by the hardware. Software can of course change the value.

The Timer 0 Comparator Value register cannot be programmed reliably by a single 64 bit write in a 32 bit environment except if only the periodic rate is being changed during run-time. If the actual Timer 0 Comparator Value needs to be reinitialized, then the following software solution will always work regardless of the environment:

- Set TIMER0_VAL_SET_CNF bit
- Set the lower 32 bits of the Timer0 Comparator Value register
- Set TIMER0_VAL_SET_CNF bit
- Set the upper 32 bits of the Timer0 Comparator Value register

Timer 0 is configurable to 32 (default) or 64 bit mode, whereas Timers 1:7 only support 32 bit mode.

WARNING

Software must be careful when programming the comparator registers. If the value written to the register is not sufficiently far in the future, then the counter may pass the value before it reaches the register and the interrupt will be missed. The BIOS should pass a data structure to the operating system to indicate that the operating system should not attempt to program the periodic timer to a rate faster than 5 us.

All of the timers support non-periodic mode.

Refer to *IA-PC HPET Specification* for more details of this mode.

Periodic Mode

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

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

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

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

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

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

Software sets the ENABLE_CNF bit to enable interrupts.

NOTE

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

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

Only timer 0 will support the periodic mode. This saves a substantial number of gates.

28.2.3.4 Enabling the Timers

The BIOS or operating system PnP code should route the interrupts. This includes the Legacy Rout bit, Interrupt Rout bit (for each timer), and interrupt type (to select the edge or level type for each timer).

The Device Driver code should do the following for an available timer:

1. Set the Overall Enable bit (Offset 10h, bit 0).
2. Set the timer type field (selects one-shot or periodic).
3. Set the interrupt enable.
4. Set the comparator value.

28.2.3.5 Interrupt Levels

Interrupts directed to the internal 8259s are active high. Refer to the **Advanced Programmable Interrupt Controller (APIC) (D31:F0)** for information regarding the polarity programming of the I/O APIC for detecting internal interrupts.

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

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

For handling interrupts and issues related to 64 bit timers with 32 bit processors, refer to IA-PC HPET Specification.

29.0 GPIO Serial Expander

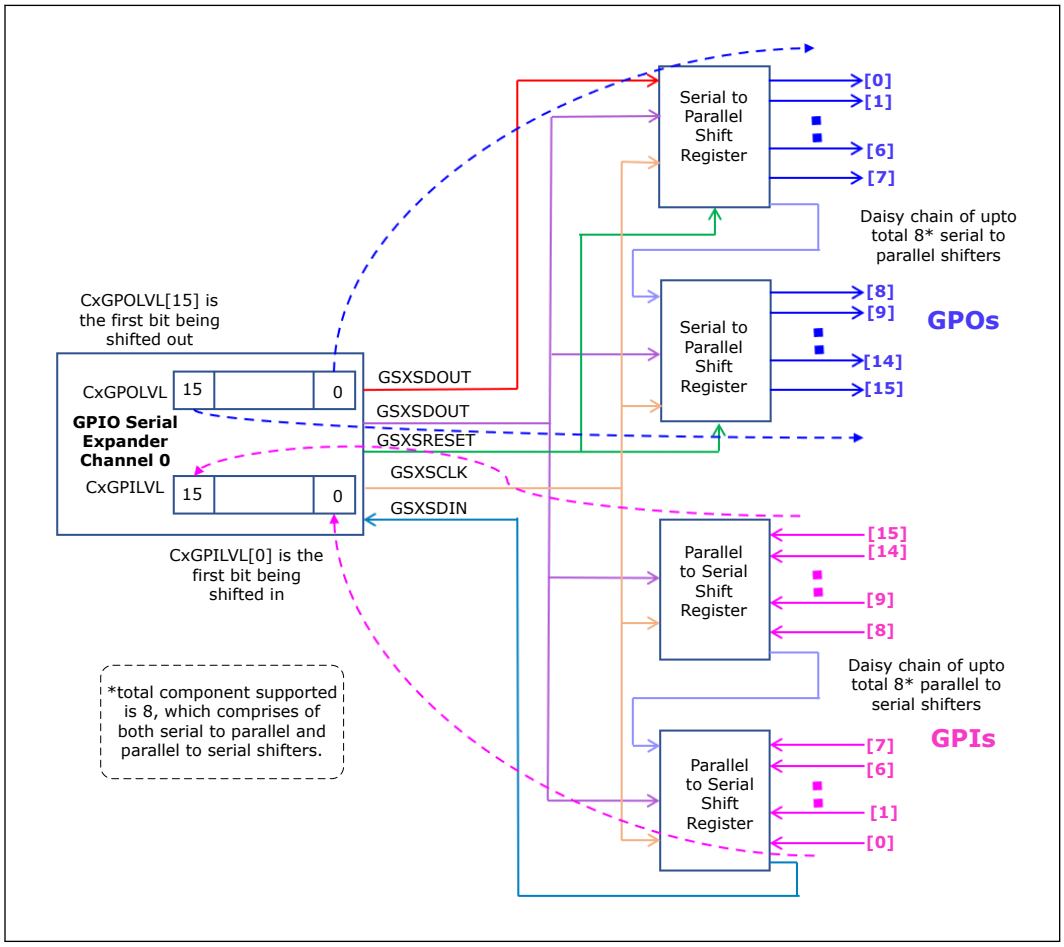
GPIO Serial Expander (GSX) is the capability provided by the Processor to expand the GPIOs on a platform that needs more GPIOs than the ones provided by the Processor. The solution requires external shift register discrete components.

29.1 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.

The below figure illustrates a GPIO expansion topology with 16 GPIs and 16 GPOs.

Figure 27. GSX Topology - Example



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.

29.2 Signal Description

Signal Name	Type	Description
GPP_F12/**GSXDOUT**/THC1_SPI2_IO0/ ISH_SPIA_MISO/GSPI1_MOSI/I2C5_SCL/USB-C_GPP_F12	O	GPIO Serial Expander Controller Data Out
GPP_F13/**GSXSLOAD**/THC1_SPI2_IO1/ ISH_SPIA_MOSI/GSPI1_MISO/I2C5_SDA/USB-C_GPP_F13	O	GPIO Serial Expander Controller Serial Load
GPP_F14/**GSXDIN**/USB-C_SMLCLK/ THC1_SPI2_IO2/GSPI0A_MOSI/USB-C_GPP_F14	I	GPIO Serial Expander Controller Data In
GPP_F15/**GSXSRESET#**/USB-C_SMLDATA/ THC1_SPI2_IO3/GSPI0A_MISO/USB-C_GPP_F15	O	GPIO Serial Expander Controller Serial Reset
GPP_F16/**GSXCLK**/THC1_SPI2_RST#/ GSPI0A_CLK/USB-C_GPP_F16	O	GPIO Serial Expander Controller Clock

29.3 Integrated Pull-ups and Pull-downs

None

30.0 Intel® Serial I/O Inter-Integrated Circuit (I^2C) Controllers

The Processor implements six I^2C controllers for six independent I^2C interfaces, I2C0-I2C5. Each interface is a two-wire serial interface consisting of a serial data line (SDA) and a serial clock (SCL).

I2C4 and I2C5 only implement the I^2C host controllers and do not incorporate a DMA controller. Therefore, I2C4 and I2C5 are restricted to operate in PIO mode only.

The I^2C interfaces support the following features:

- Speed: standard mode (up to 100 Kb/s), fast mode (up to 400 Kb/s), fast mode plus (up to 1 MB/s) and High speed mode (up to 3.2 Mb/s).
- Operate in 1.8 V only
- Host I^2C operation only
- 7-bit or 10-bit addressing
- 7-bit or 10-bit combined format transfers
- Bulk transmit mode
- Ignoring CBUS addresses (an older ancestor of I^2C used to share the I^2C bus)
- Interrupt or polled-mode operation
- Bit and byte waiting at all bus speed
- Component parameters for configurable software driver support
- Programmable SDA hold time (tHD; DAT)
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- 64-byte Tx FIFO and 64-byte Rx FIFO
- SW controlled serial data line (SDA) and serial clock (SCL)

NOTES

1. The controllers must only be programmed to operate in Host mode only. I^2C device mode is not supported.
2. I^2C multi hosts is not supported.
3. Simultaneous configuration of Fast Mode and Fast Mode Plus/High speed mode is not supported.
4. I^2C General Call is not supported.

Table 92. Acronyms

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

Table 93. References

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

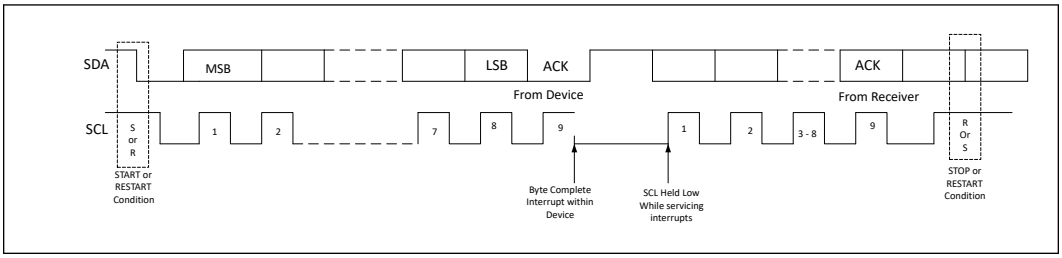
30.1 Functional Description

30.1.1 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 Host generates a START condition, signaling all devices on the bus to listen for data.
- The host writes a 7-bit address, followed by a read/write bit to select the target device and to define whether it is a transmitter or a receiver.
- The target device sends an acknowledge bit over the bus. The host must read this bit to determine whether the addressed target device is on the bus.
- Depending on the value of the read/write bit, any number of 8-bit messages can be transmitted or received by the host. These messages are specific to the 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 host with a STOP condition. This frees the bus for the next host to begin communications. When the bus is free, both data and clock lines are high.

Figure 28. Data Transfer on I²C Bus



Combined Formats

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

The Processor controllers do not support mixed address and mixed address format (which means a 7-bit address transaction followed by a 10-bit address transaction or vice versa) combined format transaction.

To initiate combined format transfers, IC_CON.IC_RESTSART_EN should be set to 1. With this value set and operating as a host, when the controller completes an 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.

30.1.2 DMA Controller

The I²C controllers 0 to 3 (I2C0 - I2C3) each has an integrated DMA controller. The I2C controller 4 and 5 (I2C4 and I2C5) only implement the I2C host controllers and do not incorporate a DMA. Therefore, I2C4 and I2C5 are restricted to operate in PIO mode only.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires the peripheral to control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires the peripheral to control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory (such as DDR or SRAM). The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode.

Channel Control

- The source transfer width and destination transfer width is programmable. The width can be programmed to 1, 2, or 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. This number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. The block size is not 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.

30.1.3 Reset

Each host controller has an independent reset associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

NOTE

To avoid a potential I^2C peripheral deadlock condition where the reset goes active in the middle of a transaction, the I^2C controller must be idle before a reset can be initiated.

30.1.4 Power Management

Device Power Down Support

To power down peripherals connected to Processor I^2C bus, the idle configured state of the I/O signals is retained to avoid voltage transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when I^2C bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The interface supports this by reporting its service latency requirements to the platform power management controller using LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

30.1.5 Interrupts

I^2C interface has an interrupt line which is used to notify the driver that service is required.

When an interrupt occurs, the device driver needs to read the host controller, DMA interrupt status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

30.1.6 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.

30.1.7 Programmable SDA Hold Time

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

30.2 Signal Description

Signal Name	Type	Description
GPP_H19/**I2C0_SDA**/I3C0_SDA/USB-C_GPP_H19	I/OD	**I²C Link 0 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H20/**I2C0_SCL**/I3C0_SCL/USB-C_GPP_H20	I/OD	**I²C Link 0 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H21/**I2C1_SDA**/I3C1_SDA/USB-C_GPP_H21	I/OD	**I²C Link 1 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H22/**I2C1_SCL**/I3C1_SCL/USB-C_GPP_H22	I/OD	**I²C Link 1 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H04/**I2C2_SDA**/ CNV_MFUART2_RXD/USB-C_GPP_H04	I/OD	**I²C Link 2 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H05/**I2C2_SCL**/ CNV_MFUART2_TXD/USB-C_GPP_H05	I/OD	**I²C Link 2 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B02/ISH_I2C0_SDA/ ISH_I3C0_SDA/**I2C2A_SDA**/USB-C_GPP_B02	I/OD	**I²C Link 2A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C2 controller, to support touch device interface convergence.
GPP_B03/ISH_I2C0_SCL/ ISH_I3C0_SCL/**I2C2A_SCL**/USB-C_GPP_B03	I/OD	**I²C Link 2A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C2 controller, to support touch device interface convergence.
GPP_H06/**I2C3_SDA**/UART1_RXD/ ISH_UART1A_RXD/USB-C_GPP_H06	I/OD	**I²C Link 3 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_H07/**I2C3_SCL**/UART1_TXD/ ISH_UART1A_TXD/USB-C_GPP_H07	I/OD	**I²C Link 3 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_D01/**I2C3A_SDA**/BKLTEN2/ ISH_I2C2A_SDA/USB-C_GPP_D01	I/OD	**I²C Link 3A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C3 controller, to support touch device interface convergence.
GPP_D02/**I2C3A_SCL**/BKLTCTL2/ ISH_I2C2A_SCL/USB-C_GPP_D02	I/OD	**I²C Link 3A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.

continued...

Signal Name	Type	Description
		Note : Alternate interface from/to the same I2C3 controller, to support touch device interface convergence.
GPP_E12/THC0_SPI1_IO1/ GSPI0_MISO/**I2C4_SDA**/USB-C_GPP_E12	I/OD	**I²C Link 4 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_E13/THC0_SPI1_IO0/ GSPI0_MOSI/**I2C4_SCL**/USB-C_GPP_E13	I/OD	**I²C Link 4 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B18/ISH_I2C2_SDA/ **I2C4A_SDA**/USB-C_GPP_B18	I/OD	**I²C Link 4A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C4 controller, to support touch device interface convergence.
GPP_B19/ISH_I2C2_SCL/**I2C4A_SCL**/ USB-C_GPP_B19	I/OD	**I²C Link 4A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C4 controller, to support touch device interface convergence.
GPP_F13/GSXSLOAD/THC1_SPI2_IO1/ ISH_SPIA_MOSI/GSPI1_MISO/ **I2C5_SDA**/USB-C_GPP_F13	I/OD	**I²C Link 5 Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_F12/GSXDOUT/THC1_SPI2_IO0/ ISH_SPIA_MISO/GSPI1_MOSI/ **I2C5_SCL**/USB-C_GPP_F12	I/OD	**I²C Link 5 Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance.
GPP_B20/**I2C5A_SDA**/ISH_GP8/USB-C_GPP_B20	I/OD	**I²C Link 5A Serial Data Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C5 controller, to support touch device interface convergence.
GPP_B21/**I2C5A_SCL**/ISH_GP9/USB-C_GPP_B21	I/OD	**I²C Link 5A Serial Clock Line** External Pull-up resistor may be required depending on Bus Capacitance. Note : Alternate interface from/to the same I2C5 controller, to support touch device interface convergence.

30.3 Integrated Pull-Ups and Pull-Downs

None.

30.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
I2C[5:0]_SDA , I2C[2:5]A_SDA	Primary	Undriven	Undriven	Undriven
I2C[5:0]_SCL , I2C[2:5]A_SCL	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

31.0 Intel® Serial I/O Improved Inter-Integrated Circuit (I3C) Controllers

I3C specification is backward compatible with I2C devices. The I3C enables dynamic address allocation and inband interrupts. The Spec also allows for hot-plug / hot-join of devices. The I3C Specification is backward compatible with legacy I2C devices and enables coexistence of legacy I2C and I3C devices on the same bus in Fast Mode, Fast Mode Plus modes, without clock stretching. The Processor has one I3C controller compliant to MIPI I3C HCI Specification, that can support 2 I3C buses and up to 8 devices per bus (subject to meeting electrical/topology requirements).

The I3C interfaces support the following features:

- Support for MIPI I3C spec v1.0, and MIPI I3C HCI Specification
- Speed: standard mode (up to 100 Kb/s), fast mode (up to 400 Kb/s), fast mode plus (up to 1 MB/s)
- Supports clock loopback using dummy IO to meet ACIO timing
- Maximum theoretical Baud rate is 12900 kbps
- Maximum validated Baud rate is 12500 kbps
- Operate in 1.8 V Only
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- 64-byte Tx FIFO and 64-byte Rx FIFO
- PME/wake support for IBI, when in S0ix
- PCI/ACPI enumeration support
- I3C static addressing and dynamic addressing support
- I3C in-band interrupt
- I3C transactions using SDR
- Error detection and recovery methods M0, M2
- For stalling Host clock on data buffering
- Host I3C operation only

NOTES

1. The controllers must only be programmed to operate in Host mode only. I3C Device mode is not supported.
2. I3C multi Hosts is not supported.
3. Simultaneous configuration of Fast Mode and Fast Mode Plus is not supported.

Table 94. Acronyms

Acronyms	Description
I^3C	Improved Inter-Integrated Circuit
SCL	Serial Clock Line
SDA	Serial Data Line

31.1 Functional Description

31.1.1 Reset

Each host controller has an independent reset associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

NOTE

To avoid a potential I^3C peripheral deadlock condition where the reset goes active in the middle of a transaction, the I^3C controller must be idle before a reset can be initiated.

31.1.2 Power Management

Device Power Down Support

To power down peripherals connected to Processor I^3C bus, the idle configured state of the I/O signals is retained to avoid voltage transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when I^3C bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The interface supports this by reporting its service latency requirements to the platform power management controller using LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements.

2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

31.1.3 Interrupts

I^3C interface has an interrupt line which is used to notify the driver that service is required.

When an interrupt occurs, the device driver needs to read the host controller, DMA interrupt status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

31.2 Signal Description

Signal Name	Type	Description
GPP_H19/I2C0_SDA/**I3C0_SDA**/USB-C_GPP_H19	I/OD	**I^3C Link 0 Serial Data Line**
GPP_H20/I2C0_SCL/**I3C0_SCL**/USB-C_GPP_H20	I/OD	**I^3C Link 0 Serial Clock Line**
GPP_H21/I2C1_SDA/**I3C1_SDA**/USB-C_GPP_H21	I/OD	**I^3C Link 1 Serial Data Line**
GPP_H22/I2C1_SCL/**I3C1_SCL**/USB-C_GPP_H22	I/OD	**I^3C Link 1 Serial Clock Line**
GPP_H10/UART0_RTS#/**I3C1A_SDA**/ISH_GP10A/USB-C_GPP_H10	I/OD	**I^3C Link 1A Serial Data Line** Note : Alternate interface from/to the same I3C1 controller, to support touch device interface convergence.
GPP_H11/UART0_CTS#/**I3C1A_SCL**/ISH_GP11A/USB-C_GPP_H11	I/OD	**I^3C Link 1A Serial Clock Line** Note : Alternate interface from/to the same I3C1 controller, to support touch device interface convergence.

31.3 Integrated Pull-Ups and Pull-Downs

None.

31.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
I3C[1:0]_SDA , I3C1A_SDA	Primary	Undriven	Undriven	Undriven
I3C[1:0]_SCL, I3C1A_SCL	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

32.0 Gigabit Ethernet Controller

The Gigabit Ethernet controller in conjunction with the Intel® Ethernet Connection I219 provides a complete LAN solution. This chapter describes the behavior of the Gigabit Ethernet Controller.

Table 95. Acronyms

Acronyms	Description
GbE	Gigabit Ethernet

Table 96. References

Specification	Location
IEEE 802.3 Ethernet	http://standards.ieee.org/getieee802/
Intel® Ethernet Connection I219 Datasheet	http://www.intel.com/content/www/us/en/

32.1 Functional Description

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

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

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

NOTE

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

NOTE

Refer the section "Flexible High Speed I/O" for GbE lane allocation options.

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

NOTE

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

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

The integrated GbE controller features are:

- Network Features
  - Compliant with the 1 GB/s Ethernet 802.3, 802.3u, 802.3ab specifications
  - Multi-speed operation: 10/100/1000 Mbps
  - Full-duplex operation at 10/100/1000 Mbps: Half-duplex at 10/100 Mbps
  - Flow control support compliant with the 802.3X specification
  - VLAN support compliant with the 802.3q specification
  - MAC address filters: perfect match unicast filters; multicast hash filtering, broadcast filter and promiscuous mode
  - PCI Express*/SMLink interface to GbE PHYs
- Host Interface Features
  - 64-bit address host support for systems using more than 4 GB of physical memory
  - Programmable host memory receive buffers (256 bytes to 16 KB)
  - Intelligent interrupt generation features to enhance driver performance
  - Descriptor ring management hardware for transmit and receive
  - Software controlled reset (resets everything except the configuration space)
  - Message Signaled Interrupts
- Performance Features
  - Configurable receive and transmit data FIFO, programmable in 1 KB increments
  - TCP segmentation off loading features
  - Fragmented UDP checksum off load for packet reassembly
  - IPv4 and IPv6 checksum off load support (receive, transmit, and large send)
  - Split header support to eliminate payload copy from user space to host space
  - Receive Side Scaling (RSS) with two hardware receive queues
  - Supports 9018 bytes of jumbo packets
  - Packet buffer size 32 KB
  - TimeSync off load compliant with 802.1as specification
  - Platform time synchronization

- Power Management Features
  - Magic Packet* wake-up enable with unique MAC address
  - ACPI register set and power down functionality supporting D0 and D3 states
  - Full wake up support (APM, ACPI)
  - MAC power down at Sx, DM-Off with and without WoL
  - Auto connect battery saver at S0 no link and Sx no link
  - Energy Efficient Ethernet (EEE) support
  - Latency Tolerance Reporting (LTR)
  - ARP and ND proxy support through LAN Connected Device proxy

32.1.1 GbE PCI Bus Interface

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

Transaction Layer

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

Data Alignment

- **4-KB Boundary**

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

- **PCI Request Size**

  PCI requests are 64 bytes or less and are aligned to make better use of memory controller resources.

Configuration Request Retry Status

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

32.1.2 Error Events and Error Reporting

Complete Abort Error Handling

A received request that violates the LAN Controller programing model will be discarded, for non posted transactions an unsuccessful completion with CA completion status will be returned.

Unsupported Request Error Handling

A received unsupported request to the LAN Controller will be discarded, for non posted transactions an unsuccessful completion with UR completion status will be returned. The URD bit will be set in ECTL register.

32.1.3 Ethernet Interface

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

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

Intel® Ethernet Connection I219

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

The integrated GbE controller supports various modes as listed in below table.

Table 97. LAN Mode Support

Mode	System State	Interface Active	Connections
Normal 10/100/1000 Mbps	S0	PCI Express*	Intel® Ethernet Connection I219
Normal 10/100/1000 Mbps	S0ix	SMLink0	
Wake-on-LAN	S0ix / Sx	SMLink0 / Wake#	
Manageability	S0ix / Sx	SMLink0	

32.1.4 PCI Power Management

The integrated GbE controller supports the Advanced Configuration and Power Interface (ACPI) specification as well as Advanced Power Management (APM). This enables the network-related activity (using an internal host wake signal) to wake up the host. For example, from S3 and S4 to S0.

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

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

- The LAN PHY is always powered when the Host and Intel® CSME systems are running.
  - SLP_LAN#='1' whenever SLP_S3#='1' or SLP_A#='1'.
- If the LAN PHY is required by Intel® CSME in Sx/M-Off , Intel® CSME must configure SLP_LAN#='1' irrespective of the power source and the destination power state. Intel® CSME must be powered at least once after G3 to configure this.
- If the LAN PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN.
- If the LAN PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN.
- If the LAN PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS.

32.2 Signal Description

Table 98. GbE LAN Signals

Signal Name	Type	Description	Availability
PCIE_5_TXP/**GbE_TXP** PCIE_5_TXN/**GbE_TXN**	O	Differential transmit pairs to the Intel® Ethernet Connection I219 based on the PCIe interface. Refer to PCI Express* (PCIe*) for details on the PCI Express*transmit signals.	H/U-Series Processor only
PCIE_5_RXP/**GbE_RXP** PCIE_5_RXN/**GbE_RXN**	I	Differential receive pairs to the Intel® Ethernet Connection I219 based on the PCIe interface. Refer to PCI Express* (PCIe*) for details on the PCI Express* transmit signals.	H/U-Series Processor only
GPP_C04/**SML0DATA**/USB-C_GPP_C04	I/OD	System Management Link data signal interface to Intel® Ethernet Connection I219. Refer to System Management Interface and SMLink for details on the SML0DATA signal. *Note:* The Intel® Ethernet Connection I219 connects to SML0DATA signal.	All Processor Series
GPP_C03/**SML0CLK**/USB-C_GPP_C03	I/OD	System Management Link data signal interface to Intel® Ethernet Connection I219. Refer to System Management Interface and SMLink for details on the SML0CLK signal. *Note:* The Intel® Ethernet Connection I219 connects to SML0CLK signal.	All Processor Series
GPP_V11/**LANPHYPC**	O	**LAN PHY Power Control:** LANPHYPC should be connected to LAN_DISABLE_N on the PHY. Processor will drive LANPHYPC low to put the PHY into a low power state when functionality is not needed. *Note:* LANPHYPC can only be driven low if SLP_LAN# is de-asserted.	H/U-Series Processor only
GPP_V12/**SLP_LAN#**	IO	(H/U-Series Processor only) **LAN Sub-System Sleep Control:** If the Gigabit Ethernet Controller is enabled, when SLP_LAN# is de-asserted it indicates that	H/U-Series Processor only

continued...

Signal Name	Type	Description	Availability
		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.	
GPP_V02/**SOC_WAKE#**	I	**SOC_WAKE:** LAN Wake Indicator from the GbE PHY. *Note:* SOC_WAKE# functionality is only supported with Intel PHY I219. Connection of a third party LAN device's wake signal to SOC_WAKE# is not supported.	H/U-Series Processor only

32.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SOC_WAKE#	External Pull-up required.	4.7 kohm +/- 5%	10 kohm +/- 5% pull-up resistor is also acceptable.

32.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5
SOC_WAKE#	Primary	Undriven	Undriven	Undriven[1]
SLP_LAN#	Primary	0	0	0/1[2]

Notes: 1. Based on wake events and Intel CSME state
2. Configurable based on BIOS settings: '0' When LAN controller is configured as "Disabled" in BIOS, SLP_LAN# will drive "Low";'1' When LAN controller is configured as "Enabled" in BIOS, SLP_LAN# will drive "High"

33.0 Connectivity Integrated (CNVi)

Connectivity Integrated (CNVi) is a general term referring to a family of connectivity solutions which are based on the Connectivity Controller family. The common component of all these solutions is the Connectivity Controller, which is embedded in the processor.

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

- The containing chip (the processor which contains the Connectivity Controller)
- Buttress (as applicable to each platform, and coupled the Connectivity Controller)
- Companion RF chip that is in a pre-certified module (i.e., M.2-2230, M.2-1216) or soldered as chip on board.

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

Table 99. Acronyms

Acronyms	Description
BRI	Bluetooth* Radio Interface
CNVi	Connectivity Integrated
SCU	System Controller Unit . It is the controller unit of CNVi.
RGI	Radio Generic interface
IP	Literally, Intellectual Property. IP refers to architecture, design, validation, and software components collectively delivered to enable one or more specific to the processor features.
MFUART	Multifunction Universal Asynchronous Receiver/Transmitter
UART	Universal Asynchronous Receiver/Transmitter

Table 100. References

Specification	Location
M.2 Specification	https://pcisig.com/specifications/pciexpress/M.2_Specification/
MIPI* Alliance specification for D-PHY v1.2	http://www.mipi.org/specifications

33.1 Functional Description

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

- **Connectivity Controller IP** contains:
  - Interfaces to the processor
  - Debug and testing interfaces
  - Power management and clock Interfaces

  - Interface to the Companion RF module (CRF)
  - Interface to physical I/O pins controlled by the processor.
  - Interfaces to the LTE modem via processor GPIO
- **Companion RF (CRF)**: This is the integrated connectivity M.2 module. The CRF Top contains:
  - Debug and testing interfaces
  - Power and clock Interfaces
  - Interface to the Connectivity Controller chip
- **Physical I/O Pins**: The SCU units are responsible for generating and controlling the power and clock resources of Connectivity Controller and CRF. There are unique SCUs in Connectivity Controller and CRF and their operation is coordinated due to power and clock dependencies. This coordination is achieved by signaling over a control bus (AUX) connecting Connectivity Controller and CRF.

Both Connectivity Controller and CRF have a dedicated AUX bus and arbiter. These two AUX buses are connected by a special interface that connects over the RGI bus. Each of the Connectivity Controller and CRF cores is dedicated to handle a specific connectivity function (Wi-Fi, Bluetooth).

Only the digital part of the connectivity function is located in Connectivity Controller cores, while the CRF cores handle some digital, but mostly analog and RF functionality. Each core in the Connectivity Controller has an interface to the host and an interface to its counterpart in CRF. CRF cores include an analog part which is connected to board level RF circuitry and to an antenna.

The H-Processor CNVi Controller supports both CNVio2 and CNVio3 also called as STEP (Serial Time Encoded Protocol) interface for Wi-Fi 7 support. The CNVio interface has two data lanes and one clock for each direction. The CNVio3 uses the same clock signals of CNVio2 and does not use the CNVio2 data signals.

CNVio3, or STEP, is a new proprietary differential interface to support Wi-Fi 7 throughput requirements. CNVio3 is a serial digital interface and operates at 12 Gbits/sec.

CNVio3 reuses CNVio2 clock traces (1 TX clock and 1 RX clock). CNVio3 maps 2 bits symbols into pulse width. CNVio3 Transmit output is a PWM signal. Each 2 bits symbols have a positive pulse and a negative pulse. The transmission starts with positive pulse, and polarity is switched for every 2 bits symbols.CNVio3 Receive decodes the pulse width back into the transmitted symbols.

33.2 Signal Description

Signal Name	Type	Description
GPIO fixed functions (Signals for Integrated Connectivity (CNVi) and Discrete Connectivity (CNVd) functions		
GPP_D14/**I2S2_SCLK**/DMIC_CLK_A0/USB-C_GPP_D14	I/O	For CNVi: Unused For discrete connectivity with UART host support: Optional Bluetooth* I^2S bus clock
GPP_F04/**CNV_RF_RESET#**/USB-C_GPP_F04	I/O	For CNVi: RF companion (CRF) reset signal, active low. Require a 75 kohm Pull-Down on platform/motherboard level. It is recommended not to use it for bootstrapping during early Platform init flows.

continued...

Signal Name	Type	Description
GPP_D16/HDA_SDI1/**I2S2_TXD**/ DMIC_CLK_B0/USB-C_GPP_D16	O	For CNVi: Unused For discrete connectivity with UART host Bluetooth* support: Optional Bluetooth* I^2S bus data output (input to Bluetooth* module)
GPP_D17/HDA_RST#/**I2S2_RXD**/ DMIC_DATA1/USB-C_GPP_D17	I	For CNVi: Unused. For discrete connectivity with UART host support: Optional Bluetooth* I^2S bus data output (input to Bluetooth* module)
GPP_F00/**CNV_BRI_DT**/UART2_RTS#/USB-C_GPP_F00	O	For CNVi: BRI bus TX. For discrete connectivity with UART host support: Bluetooth* UART RTS#
GPP_F01/**CNV_BRI_RSP**/UART2_RXD/USB-C_GPP_F01	I	For CNVi: BRI bus RX. For discrete connectivity with UART host support: Bluetooth* UART RXD
GPP_F02/**CNV_RGI_DT**/UART2_TXD/USB-C_GPP_F02	O	For CNVi: RGI bus TX. For discrete connectivity with UART host support: Bluetooth* UART TXD
GPP_F03/**CNV_RGI_RSP**/UART2_CTS#/USB-C_GPP_F03	I	For CNVi: RGI bus RX. For discrete connectivity with UART host support: Bluetooth* UART CTS#
GPP_F05/**MODEM_CLKREQ**/USB-C_GPP_F05	O	For CNVi: Processor to CRF wake indication
GPP_F06/**CNV_PA_BLANKING**/USB-C_GPP_F06	I/O	For CNVi and discrete connectivity : Optional WLAN/Bluetooth* WWAN co-existence signal. Used to be co-existence signal for external GNSS solution
GPP_H04/I2C2_SDA/**CNV_MFUART2_RXD**/ USB-C_GPP_H04	I	For CNVi and discrete connectivity: Optional WLAN/Bluetooth* WWAN co-existence signal (Input)
GPP_H05/I2C2_SCL/**CNV_MFUART2_TXD**/ USB-C_GPP_H05	O	For CNVi and discrete connectivity : Optional WLAN/Bluetooth* WWAN co-existence signal (Output)
Fixed special purpose I/O		
CNV_WT_CLKP	O	CNVio bus TX CLK+
CNV_WT_CLKN	O	CNVio bus TX CLK-
CNV_WT_D0P	O	CNVio bus Lane 0 TX+
CNV_WT_D0N	O	CNVio bus Lane 0 TX-
CNV_WT_D1P	O	CNVio bus Lane 1 TX+
CNV_WT_D1N	O	CNVio bus Lane 1 TX-
CNV_WR_CLKP	I	CNVio bus RX CLK+
CNV_WR_CLKN	I	CNVio bus RX CLK-
CNV_WR_D0P	I	CNVio bus Lane 0 RX+
CNV_WR_D0N	I	CNVio bus Lane 0 RX-
CNV_WR_D1P	I	CNVio bus Lane 1 RX+
CNV_WR_D1N	I	CNVio bus Lane 1 RX-
Selectable special purpose I/O		
U/H **USB2P_10**	I/O	Bluetooth* USB host bus (positive) for discrete connectivity. Optional to connect to a Bluetooth* USB+ pin on the Bluetooth* module. Other USB 2.0 ports can be selected for this function.

continued...

Signal Name	Type	Description
H **USB2N_10**	I/O	Bluetooth* USB host bus (negative) for discrete connectivity. Optional to connect to a Bluetooth* USB+ pin on the Bluetooth* module. Other USB 2.0 ports can be selected for this function.
PCIE_8_TXP	O	Wi-Fi* PCIe* host bus TX (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PERp0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_8_TXN	O	Wi-Fi* PCIe* host bus TX (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PERn0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_8_RXP	I	Wi-Fi* PCIe* host bus RX (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PETp0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
PCIE_8_RXN	I	Wi-Fi* PCIe* host bus RX (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* PETn0 pin on the Wi-Fi* module. Other PCIe* ports can be selected for this function.
U/H **CLKOUT_GEN4_P5**	O	Wi-Fi* PCIe* host bus clock (positive) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* REFCLKp pin on the Wi-Fi* module. Other PCIe* clocks can be selected for this function.
U//H **CLKOUT_GEN4_N5**	O	Wi-Fi* PCIe* host bus clock (negative) for discrete connectivity. Optional to connect to a Wi-Fi* PCIe* REFCLKp pin on the Wi-Fi* module. Other PCIe* clocks can be selected for this function.
CL_RST#	O	Wi-Fi* CLINK host bus reset for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK reset pin on the Intel® vPro™ Wi-Fi* module.
CL_DATA	I/O	Wi-Fi* CLINK host bus data for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK data pin on the Intel® vPro™ Wi-Fi* module.
CL_CLK	O	Wi-Fi* CLINK host bus clock for discrete connectivity with CLINK support (Intel® vPro™). Optional to connect to a Wi-Fi* CLINK clock pin on the Intel® vPro™ Wi-Fi* module.
W_Disable1# (GPIO)	O	Used for Wi-Fi* RF-Kill control. This pin can be connected to a platform switch or to processor GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init). The signal must keep value in Sx state (configured in BIOS) *Note:* Signal name not available in processor ballmap. This is a representation of GPIO used as CNVi signal.
W_Disable2# (GPIO)	O	Used for Bluetooth* RF-Kill control. This pin can be connected to a platform switch or to processor GPIOs (recommendation- if possible do not use GPIOs that have Platform impact as "bootstraps" during platform init). The signal must keep value in Sx state (configured in BIOS) *Note:* Signal name not available in processor ballmap. This is a representation of GPIO used as CNVi signal.
CNV_RCOMP	Analog	CNVi RCOMP is analog connection point for an external bias resistor(200 ohms) to ground.

33.3 Integrated Pull-ups and Pull-downs

Signal	Resistor	Value	Notes
CNV_BRI_RSP	Pull up	20 kohm	
CNV_RGI_RSP	Pull up	20 kohm	

33.4 I/O Signal Planes and States

Signal Name	Power plane	During Reset[1]	Immediately After Reset[1]	S4/S5
CNV_RF_RESET#	Primary	Undriven	Driven Low	Undriven
MODEM_CLKREQ	Primary	Driven High	Driven High	Driven High
CNV_MFUART2_RXD	Primary	Undriven	Undriven	Undriven
CNV_MFUART2_TXD	Primary	Undriven	Undriven	Undriven
CNV_BRI_DT	Primary	Driven High	Driven High	Driven High
CNV_BRI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)
CNV_RGI_DT	Primary	Driven High	Driven High	Driven High
CNV_RGI_RSP	Primary	Powered (input, PU)	Powered (input, PU)	Powered (input, PU)
CNV_WT_CLKP	Primary	Undriven	Driven Low	Undriven
CNV_WT_CLKN	Primary	Undriven	Driven Low	Undriven
CNV_WT_D0P	Primary	Undriven	Driven Low	Driven High
CNV_WT_D0N	Primary	Undriven	Driven Low	Driven High
CNV_WT_D1P	Primary	Undriven	Driven Low	Driven High
CNV_WT_D1N	Primary	Undriven	Driven Low	Driven High
CNV_WR_CLKP	Primary	Undriven	Undriven	Powered (input)
CNV_WR_CLKN	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D0P	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D0N	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D1P	Primary	Undriven	Undriven	Powered (input)
CNV_WR_D1N	Primary	Undriven	Undriven	Powered (input)
CNV_RCOMP	Primary	Undriven	Undriven	Driven High
Note: 1. Reset reference for primary well pins is RSMRST#.				

34.0 Controller Link

The controller link is used to manage the wireless devices supporting Intel® CSME Technology. Controller Link will transmit data at 60.0 Mbps on the Controller Link interface. The Controller Link clock frequency is 30.0 MHz. The Controller Link interface voltage supported is 1.25 V nominal.

NOTE

Refer to WNIC product datasheets for supported data rate and clock.

Table 101. Acronyms

Acronyms	Description
CL	Controller Link
WLAN	Wireless Local Area Network
WNIC	Wireless Network Interface Card

34.1 Signal Description

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

34.2 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value (Ohm)	Notes
CL_DATA	Pull-up Pull-down	31.25 100	I/O Signal Planes and States on page 250
CL_CLK	Pull-up Pull-down	31.25 100	

34.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[3]	Immediately After Reset[3]	S4/S5
CL_DATA	Primary	Refer to Notes	Refer to Notes	Internal Pull-down
CL_CLK	Primary	Refer to Notes	Refer to Notes	Internal Pull-down
CL_RST#	Primary	Driven Low	Driven High	Driven High

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

34.4 External CL_RST# Pin Driven/Open-drained Mode Support

The WLAN has transitioned to 1.8 V for external CL_RST# pin and the processor controller Link I/O buffer drives 1.8 V only on this pin.

35.0 Integrated Sensor Hub (ISH)

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

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

The ISH consists of the following key components:

- A combined cache for instructions and data.
  - ROM space intended for the bootloader.
  - SRAM space for code and data.
- Interfaces to sensor peripherals (I^2C, I^3C, UART, SPI, GPIO).
- An interface to main memory.
- Out of Band signals for clock and wake-up control.
- Inter Process Communications to the Host and Intel® CSME.
- Part of the PCI tree on the host.

Table 102. Acronyms

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

Table 103. References

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

35.1 Features

35.1.1 ISH I^2C Controllers

The ISH supports three I^2C controllers capable of operating at speeds up to 2.4 Mbps each. The I^2C controllers are completely independent of each other: they do not share any pins, memory spaces, or interrupts.

The ISH's I^2C host controllers share the same general specifications:

- Host I^2C operation
- Support for the following operating speeds:
  - Standard mode: 100 kbps
  - Fast Mode: 400 kbps
  - Fast Mode Plus: 1000 kbps
  - High Speed Mode: 2400 kbps
- Support for both 7-bit and 10-bit addressing formats on the I^2C bus
- FIFO of 64 bytes with programmable watermarks/thresholds

35.1.2 ISH I^3C Controllers

The ISH supports one I^3C controllers, two-Wire I^3C serial interface which consists of a Serial Data Line (SDA) and a Serial Clock (SCL).

The following are the ISH's I^3C host controller specifications:

- Modes Supported
  - I^3C Controller (Single Controller) - other modes are not supported
- In-band-interrupt (IBI) buffer depth of 16 bytes for command and status
- Supports Data transfer to legacy I^2C devices
- Clock stalling support in Controller Mode
- Supports various Data rates, such as, FM, FM+, SDR, HDR/DDR Rate
- CRC/Parity Generation and Validation
- Support for broadcast and directed CCC transfers
- Detects arbitration loss due to incoming IBI and subsequently re-transmits the command.
- Use of Duty Cycle to achieve Lower Effective Speed for SDR transfers to work with slower I^3C

35.1.3 ISH UART Controller

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

The UART has the following capabilities:

- Support for operating speeds up to 4 Mbps
- Support for auto flow control using the RTS#/CTS# signals
- 64-byte FIFO
- DMA support to allow direct transfer to the ISH local SRAM without intervention by the controller. This saves interrupts on packets that are longer than the FIFO or when there are back-to-back packets to send or receive.

35.1.4 ISH GSPI Controller

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

The SPI controller includes:

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

35.1.5 ISH GPIOs

The ISH supports eight dedicated GPIOs.

35.2 Functional Description

This section provides the information about ISH Micro-Controller, SRAM, PCI Host Interface, Power Domains and Management, ISH IPC and ISH Interrupt Handling via IOAPIC (Interrupt Controller).

35.2.1 ISH Micro-Controller

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

35.2.2 SRAM

The local SRAM is used for ISH FW code storage and to read/write operational data. The local SRAM block includes both the physical SRAM as well as the controller logic. The SRAM is a total of 640 KB organized into banks of 32 KB each and is 32-bit wide.

The SRAM is shared with Intel® CSME as shareable memory. To protect against memory errors, the SRAM includes ECC support. The ECC mechanism is able to detect multi-bit errors and correct for single bit errors. The ISH firmware has the ability to put unused SRAM banks into lower power states to reduce power consumption.

35.2.3 PCI Host Interface

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

MMIO Space

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

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

DMA Controller

The DMA controller supports up to 64-bit addressing.

PCI Interrupts

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

PCI Power Management

PME is not supported in ISH.

35.2.4 ISH IPC

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

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

Table 104. 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.

35.2.5 ISH Interrupt Handling via IOAPIC (Interrupt Controller)

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

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

35.3 Signal Description

Signal Name	Type	Description
GPP_B02/**ISH_I2C0_SDA**/**ISH_I3C0_SDA**/I2C2A_SDA/ USB-C_GPP_B02	I/OD	ISH I^2C 0 Data ISH I^3C 0 Data
GPP_B03/**ISH_I2C0_SCL**/**ISH_I3C0_SCL**/I2C2A_SCL/ USB-C_GPP_B03	I/OD	ISH I^2C 0 Clk ISH I^3C 0 Clk
GPP_H14/**ISH_UART1_RXD**/UART1A_RXD/ **ISH_I2C1_SDA**/USB-C_GPP_H14	I/OD	ISH I^2C 1 Data ISH UART1 Receive Data
GPP_H15/**ISH_UART1_TXD**/UART1A_TXD/ **ISH_I2C1_SCL**/USB-C_GPP_H15	I/OD	ISH I^2C 1 Clk ISH UART1 Transmit Data
GPP_B18/**ISH_I2C2_SDA**/I2C4A_SDA/USB-C_GPP_B18	I/OD	ISH I^2C 2 Data
GPP_B19/**ISH_I2C2_SCL**/I2C4A_SCL/USB-C_GPP_B19	I/OD	ISH I^2C 2 Clk
GPP_D01/I2C3A_SDA/BKLTEN2/**ISH_I2C2A_SDA**/USB-C_GPP_D01	I/OD	ISH I^2C 2A Data
GPP_D02/I2C3A_SCL/BKLTCTL2/**ISH_I2C2A_SCL**/USB-C_GPP_D02	I/OD	ISH I^2C 2A Clk
GPP_B05/BK1/**ISH_GP0**/SBK1/USB-C_GPP_B05	I/O	ISH GPIO 0
GPP_B06/BK2/**ISH_GP1**/SBK2/USB-C_GPP_B06	I/O	ISH GPIO 1
GPP_B07/BK3/**ISH_GP2**/SBK3/USB-C_GPP_B07	I/O	ISH GPIO 2
GPP_B08/BK4/**ISH_GP3**/SBK4/USB-C_GPP_B08	I/O	ISH GPIO 3
GPP_B04/BK0/**ISH_GP4**/SBK0/USB-C_GPP_B04	I/O	ISH GPIO 4
GPP_B22/TIME_SYNC0/**ISH_GP5**/USB-C_GPP_B22	I/O	ISH GPIO 5
GPP_B23/TIME_SYNC1/**ISH_GP6**/USB-C_GPP_B23	I/O	ISH GPIO 6
GPP_E05/SATA_DEVSLP1/**ISH_GP7**/USB-C_GPP_E05(UH)	I/O	ISH GPIO 7
GPP_B20/I2C5A_SDA/**ISH_GP8**/USB-C_GPP_B20	I/O	ISH GPIO 8
GPP_B21/I2C5A_SCL/**ISH_GP9**/USB-C_GPP_B21	I/O	ISH GPIO 9
GPP_E16/PROC_GP3/VRALERT#/**ISH_GP10**/USB-C_GPP_E16	I/O	ISH GPIO 10
GPP_F09/RSVD/SX_EXIT_HOLDOFF#/**ISH_GP11**/USB-C_GPP_F09	I/O	ISH GPIO 11
GPP_E15/PROC_GP2/RSVD/**ISH_GP5A**/USB-C_GPP_E15	I/O	ISH GPIO 5A
GPP_F10/SATAXPCIE1/SATAGP1/**ISH_GP6A**/USB-C_GPP_F10(UH)	I/O	ISH GPIO 6A

continued...

Signal Name	Type	Description
GPP_F22/**ISH_GP8A**/USB-C_GPP_F22	I/O	ISH GPIO 8A
GPP_F23/**ISH_GP9A**/USB-C_GPP_F23	I/O	ISH GPIO 9A
GPP_H10/UART0_RTS#/I3C1A_SDA/**ISH_GP10A**/USB-C_GPP_H10	I/O	ISH GPIO 10A
GPP_H11/UART0_CTS#/I3C1A_SCL/**ISH_GP11A**/USB-C_GPP_H11	I/O	ISH GPIO 11A
GPP_D06/**ISH_UART0_TXD**/**ISH_SPI_CLK**/SML0BCLK/USB-C_GPP_D06	O	ISH UART 0 Transmit Data ISH SPI Clock
GPP_D05/**ISH_UART0_RXD**/**ISH_SPI_CS#**/SML0BDATA/USB-C_GPP_D05	I	ISH UART 0 Receive Data ISH SPI Chip Select
GPP_D07/IMGCLKOUT4/**ISH_UART0_RTS#**/**ISH_SPI_MISO**/USB-C_GPP_D07	O	ISH UART 0 Request To Send ISH SPI MISO
GPP_D08/**ISH_UART0_CTS#**/**ISH_SPI_MOSI**/SML0BALERT#/USB-C_GPP_D08	I	ISH UART 0 Clear to Send ISH SPI MOSI
GPP_H07/I2C3_SCL/UART1_TXD/**ISH_UART1A_TXD**/USB-C_GPP_H07	O	ISH UART 1A Transmit Data
GPP_H06/I2C3_SDA/UART1_RXD/**ISH_UART1A_RXD**/USB-C_GPP_H06	I	ISH UART 1A Receive Data
GPP_F17/THC1_SPI2_CS#/**ISH_SPIA_CS#**/GSPI1_CS0#/USB-C_GPP_F17	O	ISH SPIA Chip Select
GPP_F11/THC1_SPI2_CLK/**ISH_SPIA_CLK**/GSPI1_CLK/USB-C_GPP_F11	O	ISH SPIA Clock
GPP_F12/GSXDOUT/THC1_SPI2_IO0/**ISH_SPIA_MISO**/GSPI1_MOSI/I2C5_SCL/USB-C_GPP_F12	I	ISH SPIA MISO
GPP_F13/GSXSLOAD/THC1_SPI2_IO1/**ISH_SPIA_MOSI**/GSPI1_MISO/I2C5_SDA/USB-C_GPP_F13	O	ISH SPIA MOSI

35.4 Integrated Pull-Ups and Pull-Down

NA

35.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ISH_I2C0_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C0_SCL	Primary	Undriven	Undriven	Undriven
ISH_I2C1_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C1_SCL	Primary	Undriven	Undriven	Undriven
ISH_I2C2_SDA	Primary	Undriven	Undriven	Undriven
ISH_I2C2_SCL	Primary	Undriven	Undriven	Undriven
ISH_I3C0_SDA	Primary	Undriven	Undriven	Undriven
ISH_I3C0_SCL	Primary	Undriven	Undriven	Undriven

continued...

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ISH_GP[11:0]	Primary	Undriven	Undriven	Undriven
ISH_GP[11:8]A ISH_GP[6:5]A	Primary	Undriven	Undriven	Undriven
ISH_UART0_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART0_RXD	Primary	Undriven	Undriven	Undriven
ISH_UART0_RTS#	Primary	Undriven	Undriven	Undriven
ISH_UART0_CTS#	Primary	Undriven	Undriven	Undriven
ISH_UART1_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART1_RXD	Primary	Undriven	Undriven	Undriven
ISH_UART1A_TXD	Primary	Undriven	Undriven	Undriven
ISH_UART1A_RXD	Primary	Undriven	Undriven	Undriven
ISH_SPI_CS#	Primary	Undriven	Undriven	Undriven
ISH_SPI_CLK	Primary	Undriven	Undriven	Undriven
ISH_SPI_MISO	Primary	Undriven	Undriven	Undriven
ISH_SPI_MOSI	Primary	Undriven	Undriven	Undriven
ISH_SPIA_CS#	Primary	Undriven	Undriven	Undriven
ISH_SPIA_CLK	Primary	Undriven	Undriven	Undriven
ISH_SPIA_MISO	Primary	Undriven	Undriven	Undriven
ISH_SPIA_MOSI	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

36.0 System Management Interface and SMLink

The Processor provides two SMLink interfaces, SMLink0 and SMLink1. The interfaces are intended for system management and are controlled by the Intel® CSME. Refer to System Management on page 260 for more detail.

Table 105. Acronyms

Acronyms	Description
BMC	Baseboard Management Controller
EC	Embedded Controller

36.1 Functional Description

The SMLink interfaces are controlled by the Intel® CSME.

SMLink0 is mainly used for integrated LAN. When an Intel LAN PHY is connected to SMLink0, a SMT3_EN soft strap must be set to indicate that the PHY is connected to SMLink0. The interface will be running at the frequency of up to 1 MHz depending on different factors such as board routing or bus loading when the Fast Mode is enabled using a soft strap.

SMLink1 can be used with an Embedded Controller (EC) or Baseboard Management Controller (BMC).

Both SMLink0 and SMLink1 support up to 1 MHz.

36.1.1 Integrated USB-C* Usage

SMLink1 is used to communicate with USB-C* PD Controller on the platform to configure different modes such as USB, DP, Thunderbolt etc. When used for Integrated USB-C* purposes, a soft strap must be set to indicate that integrated USB-C ports from Processor are being used.

SMLINK1 uses controller mode and gets an alert signal from PMCALERT#.

Based on capabilities of different PD Controllers, re-timers needed for USB-C* connector on the platform may need to be controlled by the processor also. In these cases, both PD Controller and Re-timers will be connected to SMLink1. SMLink1 is used for all USB-C connectors on the platform.

U/H-SKU supports four integrated USB-C ports.Due to this, there could be maximum of four PD Controller and four re-timers. This translates to maximum of eight devices on the SMLINK1 bus for a platform.

USB-C* connectors are present at one side or both side of the system,so (SMLink1, PMCAlert) could be routed to long distance on the motherboard provided total bus capacitance specification is met.

USB-C* Re-timer control (like Firmware Load, USB-C configuration) handling depends on the number of I^2C ports available on the PD controller.

If the PD controller has two I^2C ports then Processor PMC will handle the Re-timer and PD controller, but if the PD controller has three or more I^2C ports then Processor PMC will handle only PD controller. Re-timers can be handled by PD controller.

SMLink1 should be run at 400 kHz when used for USB-C* purposes.

36.2 Signal Description

Signal Name	Type	Description
GPP_C04/**SML0DATA**/ USB-C_GPP_C04	I/OD	**System Management Link 0 Data**: SMBus link to external PHY. External Pull-up resistor required.
GPP_C03/**SML0CLK**/ USB-C_GPP_C03	I/OD	**System Management Link 0 Clock** External Pull-up resistor required.
GPP_C05/ **SML0ALERT#**/USB-C_GPP_C05	I/OD	**System Management 0 Alert**: Alert for the SMBus controller to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_C06/**SML1CLK**/ USB-C_GPP_C06	I/OD	**System Management Link 1 Clock**: SMBus link to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_C07/**SML1DATA**/ USB-C_GPP_C07	I/OD	**System Management Link 1 Data**: SMBus link to optional Embedded Controller or BMC. External Pull-up resistor required.
GPP_C08/ **SML1ALERT#**/ SOCHOT#/USB-C_GPP_C08	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 SOCHOT# usage. This is **NOT** the right Alert pin for USB-C* usage. External Pull-up resistor is required on this pin.
GPP_F15/GSXSRESET#/ **USB-C_SMLDATA**/ THC1_SPI2_IO3/ GSPI0A_MISO/USB-C_GPP_F15	I/OD	**System Management** bus over **Sideband 2 Core** Data External Pull-up resistor required.
GPP_F14/GSXDIN/**USB-C_SMLCLK**/ THC1_SPI2_IO2/ GSPI0A_MOSI/USB-C_GPP_F14	I/OD	**System Management** bus over **Sideband 2 Core** Clock External Pull-up resistor required.
GPP_E02/**USB-C_SMLADATA**/ THC0_SPI1_IO3/USB-C_GPP_E02	I/OD	**System Management** bus over **Sideband 2 Core** Data External Pull-up resistor required.
GPP_E01/**USB-C_SMLACLK**/ THC0_SPI1_IO2/USB-C_GPP_E01	I/OD	**System Management** bus over **Sideband 2 Core** Clock External Pull-up resistor required.
INTRUDER#	I	**Intruder Detect.**

36.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SML[4:0, 0B]ALERT#	Pull-down	20 kohm ± 30%	The internal pull-down resistor is disable after RSMRST# de-asserted.

36.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
INTRUDER#	RTC	Undriven	Undriven	Undriven
SML[4:0, 0B]DATA	Primary	Undriven	Undriven	Undriven
SML[4:0, 0B]CLK	Primary	Undriven	Undriven	Undriven
SML[4:0, 0B]ALERT#	Primary	Pull-down (Internal)	Driven Low	Pull-down (Internal)
SOCHOT#	Primary	Pull-down (Internal)	Driven Low	Pull-down (Internal)
Note: 1. Reset reference for primary well pin is RSMRST# and RTC well pin is RTCRST#.				

36.5 System Management

The Processor provides various functions to make a system easier to manage and to lower the Total Cost of Ownership (TCO) of the system. Features and functions can be augmented using external A/D converters and GPIOs, as well as an external micro controller.

The following features and functions are supported:

- First timer timeout to generate SMI# after programmable time:
  - The first timer timeout causes a SMI#, allowing SMM-based recovery from OS lock up
- Second hard-coded timer timeout to generate reboot:
  - This second timer is used only after the 1st timeout occurs
  - The second timeout allows for automatic system reset and reboot if a HW error is detected
  - Option to prevent reset the second timeout
- 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#.

Table 106. Acronyms

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

36.5.1 Theory of Operation

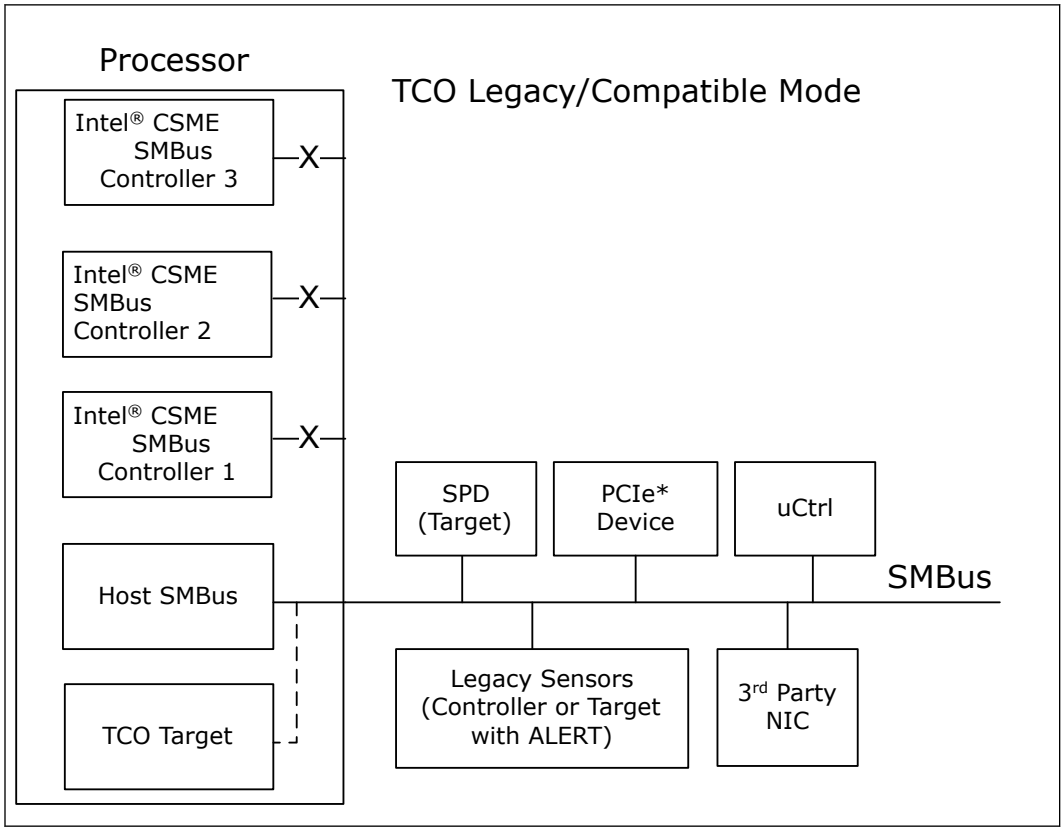
The System Management functions are designed to allow the system to diagnose failing subsystems. The intent of this logic is that some of the system management functionality can be provided without the aid of an external microcontroller.

36.5.1.1 TCO Modes

TCO Compatible Mode

In TCO Legacy/Compatible mode, only the host SMBus is used. The TCO target is connected to the host SMBus internally by default. In this mode, the Intel® Management Engine (Intel® CSME) SMBus controllers are not used and should be disabled by soft strap.

Figure 29. TCO Compatible Mode SMBus Configuration



In TCO Legacy/Compatible mode the Processor can function directly with an external LAN controller or equivalent external LAN controller to report messages to a network management console. The table below includes a list of events that will report messages to the network management console.

Table 107. Event Transitions that Cause Messages

Event	Assertion?	Deassertion?	Comments
INTRUDER# pin	Yes	No	System must hung in S0 state
Watchdog Timer Expired	Yes	NA	System will enter to hung state
SMBALERT# pin	Yes	Yes	System must hung in S0 state
BATLOW#	Yes	Yes	System must hung in S0 state
SYSPWR_FLR	Yes	No	System will enter to hung state

Advanced TCO Mode

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

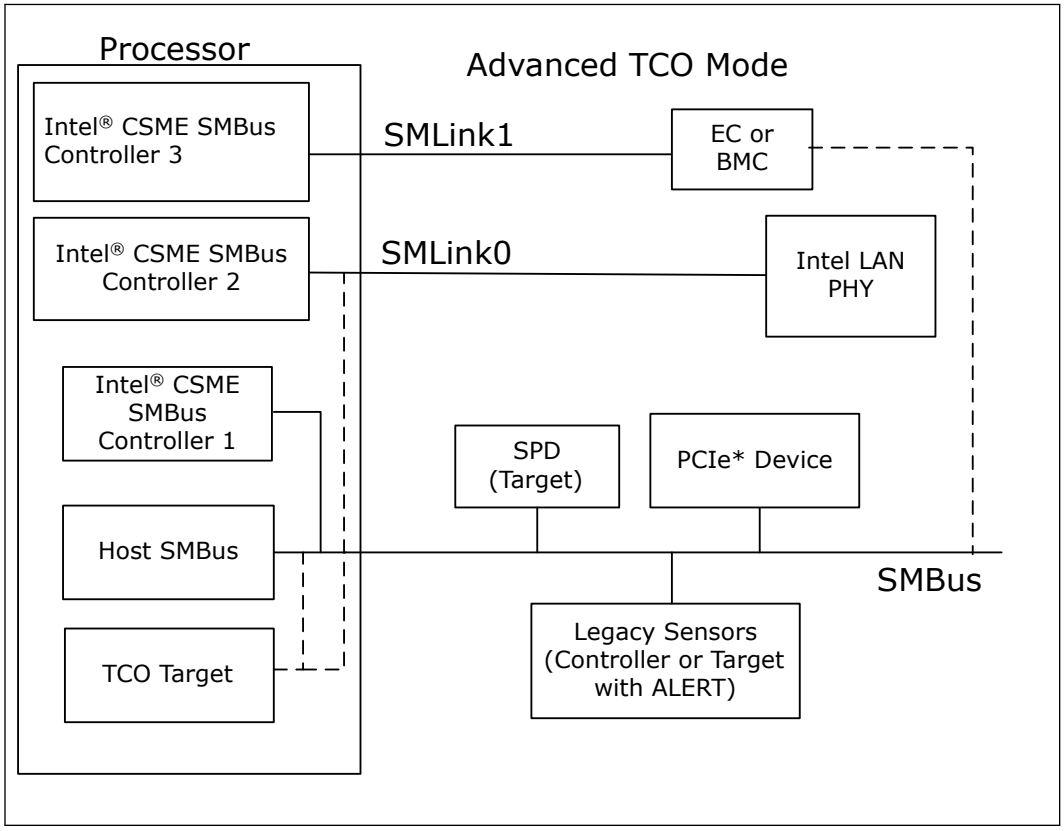
In this mode, the Intel® CSME SMBus controllers must be enabled by soft strap in the flash descriptor. Refer to figure below for more details.

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

SMLink0 is targeted 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. 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 Management Engine through the Intel® CSME SMBus connected to SMLink1. The host and TCO target communicate with BMC through SMBus.

Figure 30. Advanced TCO Mode



37.0 Host System Management Bus (SMBus) Controller

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

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

Table 108. Acronyms

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

Table 109. References

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

37.1 Functional Description

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

- **Host Controller:** Provides a mechanism for the processor to initiate communications with SMBus peripherals (Devices). The Processor is also capable of operating in a mode in which it can communicate with I^2C compatible devices.
- **Target Interface:** Allows an external host to read from or write to the Processor . Write cycles can be used to cause certain events or pass messages, and the read cycles can be used to determine the state of various status bits. The Processor ’s internal host controller cannot access the Processor ’s internal Device Interface.

37.1.1 Host Controller

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

The Processor can perform SMBus messages with either Packet Error Checking (PEC) enabled or disabled. The actual PEC calculation and checking is performed in SW. The SMBus host controller logic can automatically append the CRC byte if configured to do so.

The SMBus Address Resolution Protocol (ARP) is supported by using the existing host controller commands through software, except for the Host Notify command (which is actually a received message).

The Processor SMBus host controller checks for parity errors as a target. If an error is detected, the detected parity error bit in the PCI Status Register is set.

Host Controller Operation Overview

The SMBus host controller is used to send commands to other SMBus Target devices. Software sets up the host controller with an address, command, and, for writes, data and optional PEC; and then tells the controller to start. When the controller has finished transmitting data on writes, or receiving data on reads, it generates an SMI# or interrupt, if enabled.

The host controller supports eight command protocols of the SMBus interface (refer to the System Management Bus (SMBus) Specification, Version 2.0): Quick Command, Send Byte, Receive Byte, Write Byte/Word, Read Byte/Word, Process Call, Block Read/Write, and Block Write–Block Read Process Call.

The SMBus host controller requires that the various data and command fields be setup for the type of command to be sent. When software sets the START bit, the SMBus Host controller performs the requested transaction, and interrupts the processor (or generates an SMI#) when the transaction is completed. Once a START command has been issued, the values of the "active registers" (Host Control, Host Command, Transmit Target Address, Data 0, Data 1) should not be changed or read until the interrupt status message (INTR) has been set (indicating the completion of the command). Any register values needed for computation purposes should be saved prior to issuing of a new command, as the SMBus host controller updates all registers while completing the new command.

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

Using the SMB host controller to send commands to the Processor SMB Target port is not supported.

Command Protocols

In all of the following commands, the Host Status Register (offset 00h) is used to determine the progress of the command. While the command is in operation, the HOST_BUSY bit is set. If the command completes successfully, the INTR bit will be set in the Host Status Register. If the device does not respond with an acknowledge, and the transaction times out, the DEV_ERR bit is set.

If software sets the KILL bit in the Host Control Register while the command is running, the transaction will stop and the FAILED bit will be set after the Processor forces a time - out. In addition, if KILL bit is set during the CRC cycle, both the CRCE and DEV_ERR bits will also be set.

Quick Command

When programmed for a Quick Command, the Transmit Target Address Register is sent. The PEC byte is never appended to the Quick Protocol. Software should force the PEC_EN bit to 0 when performing the Quick Command. Software must force the I2C_EN bit to 0 when running this command. Refer to Section 5.5.1 of the System Management Bus (SMBus) Specification, Version 2.0 for the format of the protocol.

Send Byte/Receive Byte

For the Send Byte command, the Transmit Target Address and Device Command Registers are sent. For the Receive Byte command, the Transmit Target Address Register is sent. The data received is stored in the DATA0 register. Software must force the I2C_EN bit to 0 when running this command.

The Receive Byte is similar to a Send Byte, the only difference is the direction of data transfer. Refer to Sections 5.5.2 and 5.5.3 of the System Management Bus *(SMBus)* Specification, Version 2.0 for the format of the protocol.

Write Byte/Word

The first byte of a Write Byte/Word access is the command code. The next 1 or 2 bytes are the data to be written. When programmed for a Write Byte/Word command, the Transmit Target Address, Device Command, and Data0 Registers are sent. In addition, the Data1 Register is sent on a Write Word command. Software must force the I2C_EN bit to 0 when running this command. Refer to Section 5.5.4 of the *System Management Bus (SMBus) Specification*, Version 2.0 for the format of the protocol.

Read Byte/Word

Reading data is slightly more complicated than writing data. First the Processor must write a command to the Target device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The target then returns 1 or 2 bytes of data. Software must force the I2C_EN bit to 0 when running this command.

When programmed for the read byte/word command, the Transmit Target Address and Device Command Registers are sent. Data is received into the DATA0 on the read byte, and the DAT0 and DATA1 registers on the read word. Refer to Section 5.5.5 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Process Call

The process call is so named because a command sends data and waits for the target to return a value dependent on that data. The protocol is simply a Write Word followed by a Read Word, but without a second command or stop condition.

When programmed for the Process Call command, the Processor transmits the Transmit Target Address, Host Command, DATA0 and DATA1 registers. Data received from the device is stored in the DATA0 and DATA1 registers.

The Process Call command with I2C_EN set and the PEC_EN bit set produces undefined results. Software must force either I2C_EN or PEC_EN to 0 when running this command. Refer to Section 5.5.6 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

NOTES

1. For process call command, the value written into bit 0 of the Transmit Target Address Register needs to be 0.
2. If the I2C_EN bit is set, the protocol sequence changes slightly, the Command Code (Bits 18:11 in the bit sequence) are not sent. As a result, the target will not acknowledge (Bit 19 in the sequence).

Block Read/Write

The Processor contains a 32 - byte buffer for read and write data which can be enabled by setting bit 1 of the Auxiliary Control register at offset 0Dh in I/O space, as opposed to a single byte of buffering. This 32 - byte buffer is filled with write data before transmission, and filled with read data on reception. In the Processor , the interrupt is generated only after a transmission or reception of 32 bytes, or when the entire byte count has been transmitted/received.

The byte count field is transmitted but ignored by the Processor as software will end the transfer after all bytes it cares about have been sent or received.

For a Block Write, software must either force the I2C_EN bit or both the PEC_EN and AAC bits to 0 when running this command.

The block write begins with a target address and a write condition. After the command code the Processor issues a byte count describing how many more bytes will follow in the message. If a target had 20 bytes to send, the first byte would be the number 20 (14h), followed by 20 bytes of data. The byte count may not be 0. A Block Read or Write is allowed to transfer a maximum of 32 data bytes.

When programmed for a block write command, the Transmit target Address, Device Command, and Data0 (count) registers are sent. Data is then sent from the Block Data Byte register; the total data sent being the value stored in the Data0 Register.

On block read commands, the first byte received is stored in the Data0 register, and the remaining bytes are stored in the Block Data Byte register. Refer to section 5.5.7 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

NOTE

For Block Write, if the I2C_EN bit is set, the format of the command changes slightly. The Processor will still send the number of bytes (on writes) or receive the number of bytes (on reads) indicated in the DATA0 register. However, it will not send the contents of the DATA0 register as part of the message. When operating in I^2C mode (I2C_EN bit is set), the Processor will never use the 32 - byte buffer for any block commands.

I^2C* Read

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

However, this does not allow access to devices using the I^2C “Combined Format” that has data bytes after the address. Typically these data bytes correspond to an offset (address) within the serial memory chips.

NOTE

This command is supported independent of the setting of the I2C_EN bit. The I^2C Read command with the PEC_EN bit set produces undefined results. Software must force both the PEC_EN and AAC bit to 0 when running this command.

For I^2C Read command, the value written into bit 0 of the Transmit Target Address Register (SMB I/O register, offset 04h) needs to be 0.

The format that is used for the command is shown in the table below:

Table 110. I²C* Block Read

Bit	Description
1	Start
8:2	Target Address – 7 bits
9	Write
10	Acknowledge from target
18:11	Send DATA1 register
19	Acknowledge from target
20	Repeated Start
27:21	Target Address – 7 bits
28	Read
29	Acknowledge from target
37:30	Data byte 1 from target – 8 bits
38	Acknowledge
46:39	Data byte 2 from target – 8 bits
47	Acknowledge
–	Data bytes from target / Acknowledge
–	Data byte N from target – 8 bits
–	NOT Acknowledge
–	Stop

The Processor will continue reading data from the peripheral until the NAK is received.

Block Write – Block Read Process Call

The block write - block read process call is a two - part message. The call begins with a target address and a write condition. After the command code the host issues a write byte count (M) that describes how many more bytes will be written in the first part of the message. If a controller has 6 bytes to send, the byte count field will have the value 6 (0000 0110b), followed by the 6 bytes of data. The write byte count (M) cannot be 0.

The second part of the message is a block of read data beginning with a repeated start condition followed by the target address and a Read bit. The next byte is the read byte count (N), which may differ from the write byte count (M). The read byte count (N) cannot be 0.

The combined data payload must not exceed 32 bytes. The byte length restrictions of this process call are summarized as follows:

- $M \geq 1$ 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 target address and using the normal PEC computational rules. It is highly recommended that a PEC byte be used with the Block Write - Block Read Process Call. Software must do a read to the command register (offset 2h) to reset the 32 byte buffer pointer prior to reading the block data register.

NOTES

1. There is no STOP condition before the repeated START condition, and that a NACK signifies the end of the read transfer.
2. E32B bit in the Auxiliary Control register must be set when using this protocol.

Refer to Section 5.5.8 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Bus Arbitration

Several controllers may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The Processor continuously monitors the SMBDATA line. When the Processor is attempting to drive the bus to a 1 by letting go of the SMBDATA line, and it samples SMBDATA low, then some other controller is driving the bus and the Processor will stop transferring data.

If the Processor detects that it has lost arbitration, the condition is called a collision. The Processor will set the BUS_ERR bit in the Host Status Register, and if enabled, generates an interrupt or SMI#. The processor is responsible for restarting the transaction.

Clock Stretching

Some devices may not be able to handle their clock toggling at the rate that the Processor as an SMBus controller would like. They have the capability of stretching the low time of the clock. When the Processor attempts to release the clock (allowing the clock to go high), the clock will remain low for an extended period of time.

The Processor monitors the SMBus clock line after it releases the bus to determine whether to enable the counter for the high time of the clock. While the bus is still low, the high time counter must not be enabled. Similarly, the low period of the clock can be stretched by an SMBus controller if it is not ready to send or receive data.

Bus Timeout (Processor as SMBus controller)

If there is an error in the transaction, such that an SMBus device does not signal an acknowledge or holds the clock lower than the allowed Timeout time, the transaction will time out. The Processor will discard the cycle and set the DEV_ERR bit. The timeout minimum is 25 ms (800 RTC clocks). The Timeout counter inside the Processor will start after the first bit of data is transferred by the Processor and it is waiting for a response.

The 25 - ms Timeout counter will not count under the following conditions:

1. BYTE_DONE_STATUS bit (SMBus I/O Offset 00h, Bit 7) is set
2. The SECOND_TO_STS bit (TCO I/O Offset 06h, Bit 1) is not set (this indicates that the system has not locked up).

Interrupts/SMI#

The Processor SMBus controller uses PIRQB# as its interrupt pin. However, the system can alternatively be set up to generate SMI# instead of an interrupt, by setting the SMBUS_SMI_EN bit.

The three tables below, specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and target SMI, and Wake internal signals. The rows in the tables are additive, which means that if more than one row is true for a particular scenario then the Results for all of the activated rows will occur.

Table 111. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	SMBALERT_DIS (Target Command I/O Register, Offset 11h, Bit 2)	Result
SMBALERT# asserted low (always reported in Host Status Register, Bit 5)	X	X	X	Wake generated
	X	1	0	Target SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 112. Enables for SMBus Target Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	Event
Target Write to Wake/ SMI# Command	X	X	Wake generated when asleep. Target SMI# generated when awake (SMBUS_SMI_STS).
Target Write to SMLINK_SLAVE_SMI Command	X	X	Target SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

Table 113. Enables for the Host Notify Command

HOST_NOTIFY_INTREN (Target Control I/O Register, Offset 11h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Target Control I/O Register, Offset 11h, Bit 1)	Result
0	X	0	None
X	X	1	Wake generated
1	0	X	Interrupt generated
1	1	X	Target SMI# generated (SMBUS_SMI_STS)

SMBus CRC Generation and Checking

If the AAC bit is set in the Auxiliary Control register, the Processor automatically calculates and drives CRC at the end of the transmitted packet for write cycles, and will check the CRC for read cycles. It will not transmit the contents of the PEC register for CRC. The PEC bit must not be set in the Host Control register if this bit is set, or unspecified behavior will result.

If the read cycle results in a CRC error, the DEV_ERR bit and the CRCE bit in the Auxiliary Status register at Offset 0Ch will be set.

37.1.2 SMBus Target Interface

The Processor SMBus Target interface is accessed using the SMBus. The SMBus target logic will not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol device. The target interface allows the Processor to decode cycles, and allows an external micro controller to perform specific actions.

Key features and capabilities include:

- Supports decode of three types of messages: Byte Write, Byte Read, and Host Notify.
- Receive Target Address register: This is the address that the Processor decodes. A default value is provided so that the target interface can be used without the processor having to program this register.
- Receive Target Data register in the SMBus I/O space that includes the data written by the external micro controller.
- Registers that the external micro controller can read to get the state of the Processor .
  - Status bits to indicate that the SMBus target logic caused an interrupt or SMI# Bit 0 of the target Status Register for the Host Notify command.
  - Bit 16 of the SMI Status Register for all others.

NOTE

The external micro controller should not attempt to access the Processor SMBus target logic until either:

- 800 milliseconds after both: RTCRST# is high and RSMRST# is high, OR
- The PLTRST# de - asserts

If a controller leaves the clock and data bits of the SMBus interface at 1 for 50 µs or more in the middle of a cycle, the Processor target logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the target logic.

Format of Target Write Cycle

The external controller performs Byte Write commands to the Processor SMBus Target I/F. The “Command” field (bits 11:18) indicate which register is being accessed. The Data field (bits 20:27) indicate the value that should be written to that register.

The table below has the values associated with the registers.

Table 114. Target Write Registers

Register	Function
0	Command Register. Refer to the table below for valid values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–FFh	Reserved
Note: The external micro controller is responsible to make sure that it does not update the contents of the data byte registers until they have been read by the system processor. The Processor overwrites the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. The Processor will not attempt to cover this race condition (that is, unpredictable results in this case).	

Table 115. Command Types

Command Type	Description
0	Reserved
1	**WAKE/SMI#.** This command wakes the system if it is not already awake. If system is already awake, an SMI# is generated.
2	**Unconditional Powerdown.** This command sets the PWRBTNOR_STS bit, and has the same effect as the Power button Override occurring.
3	**HARD RESET WITHOUT CYCLING:** This command causes a soft reset of the system (does not include cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 2:1 set to 1, but Bit 3 set to 0.
4	**HARD RESET SYSTEM.** This command causes a hard reset of the system (including cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 3:1 set to 1.
5	**Disable the TCO Messages.** This command will disable the Processor from sending Heartbeat and Event messages. Once this command has been executed, Heartbeat and Event message reporting can only be re - enabled by assertion and then de - assertion of the RSMRST# signal.
6	**WD RELOAD:** Reload watchdog timer.
7	Reserved
8	**SMLINK_SLV_SMI.** When the Processor detects this command type while in the S0 state, it sets the SMLINK_SLV_SMI_STS bit. This command should only be used if the system is in an S0 state. If the message is received during S4 and S5 states, the Processor acknowledges it, but the SMLINK_SLV_SMI_STS bit does not get set. *Note:* It is possible that the system transitions out of the S0 state at the same time that the SMLINK_SLV_SMI command is received. In this case, the SMLINK_SLV_SMI_STS bit may get set but not serviced before the system goes to sleep. Once the system returns to S0, the SMI associated with this bit would then be generated. Software must be able to handle this scenario.
9–FFh	Reserved.

Format of Read Command

The external controller performs Byte Read commands to the Processor SMBus Target interface. The “Command” field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 116. Target Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Micro controller	
2–8	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
9	Write	External Micro controller	Always 0
10	ACK	Processor	
11–18	Command code – 8 bits	External Micro controller	Indicates which register is being accessed. Refer to the Table below for a list of implemented registers.
19	ACK	Processor	
20	Repeated Start	External Micro controller	
21–27	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
28	Read	External Micro controller	Always 1
29	ACK	Processor	
30–37	Data Byte	Processor	Value depends on register being accessed. Refer to the Table below for a list of implemented registers.
38	NOT ACK	External Micro controller	
39	Stop	External Micro controller	

Table 117. Data Values for Target Read Registers

Register	Bits	Description
0	7:0	Reserved
1	2:0	**System Power State** 000 = S0 100 = S4 101 = S5 Others = Reserved
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** *Note:* The Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the Processor will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect**. 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	Reserved
	2	Reserved
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.

continued...

Register	Bits	Description
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status.** Reflects the value of the SMBALERT# pin (when the pin is configured to SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always returns 1 if SMBALERT_DISABLE = 1.
5	0	Reserved
	1	**Battery Low Status.** 1 if the BATLOW# pin a low.
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD:** Indicates the failure core power well ramp during boot/resume. This bit will be active if the SLP_S3# pin is de - asserted and PLT_PWROK pin is not asserted.
	6	**Thermal Trip:** This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message
	7	Reserved: Default value is "X" *Note:* Software should not expect a consistent value when this bit is read through SMBUS/ SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC
E	7:0	Month of the RTC
F	7:0	Year of the RTC
10h–FFh	7:0	Reserved

- **Behavioral Notes**

  According to SMBus protocol, Read and Write messages always begin with a Start bit—Address—Write bit sequence. When the Processor detects that the address matches the value in the Receive target Address register, it will assume that the protocol is always followed and ignore the Write bit (Bit 9) and signal an Acknowledge during bit 10. In other words, if a Start—Address—Read occurs (which is invalid for SMBus Read or Write protocol), and the address matches the Processor 's Target Address, the Processor will still grab the cycle.

  Also according to SMBus protocol, a Read cycle contains a Repeated Start—Address—Read sequence beginning at Bit 20. Once again, if the Address matches the Processor 's Receive Target Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Target Read cycle.

Target Read of RTC Time Bytes

The Processor SMBus target interface allows external SMBus controller to read the internal RTC's time byte registers.

The RTC time bytes are internally latched by the Processor's hardware whenever RTC time is not changing and SMBus is idle. This ensures that the time byte delivered to the target read is always valid and it does not change when the read is still in progress on the bus. The RTC time will change whenever hardware update is in progress, or there is a software write to the RTC time bytes.

The Processor SMBus Target interface only supports Byte Read operation. The external SMBus controller will read the RTC time bytes one after another. It is the software's responsibility to check and manage the possible time rollover when subsequent time bytes are read.

For example, assuming the RTC time is 11 hours: 59 minutes: 59 seconds. When the external SMBus controller reads the hour as 11, then proceeds to read the minute, it is possible that the rollover happens between the reads and the minute is read as 0. This results in 11 hours: 0 minute instead of the correct time of 12 hours: 0 minutes. Unless it is certain that rollover will not occur, software is required to detect the possible time rollover by reading multiple times such that the read time bytes can be adjusted accordingly if needed.

Format of Host Notify Command

The Processor tracks and responds to the standard Host Notify command as specified in the *System Management Bus (SMBus) Specification,* Version 2.0. The host address for this command is fixed to 0001000b. If the Processor already has data for a previously - received host notify command which has not been serviced yet by the host software (as indicated by the HOST_NOTIFY_STS bit), then it will NACK following the host address byte of the protocol. This allows the host to communicate non - acceptance to the controller and retain the host notify address and data values for the previous cycle until host software completely services the interrupt.

NOTE

Host software must always clear the HOST_NOTIFY_STS bit after completing any necessary reads of the address and data registers.

The table below shows the Host Notify format:

Table 118. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Controller	
8:2	SMB Host Address – 7 bits	External Controller	Always 0001_000
9	Write	External Controller	Always 0
10	ACK (or NACK)	Processor	Processor NACKs if HOST_NOTIFY_STS is 1
17:11	Device Address – 7 bits	External Controller	Indicates the address of the controller ; loaded into the Notify Device Address Register
18	Unused – Always 0	External Controller	7 - bit - only address; this bit is inserted to complete the byte

continued...

Bit	Description	Driven By	Comment
19	ACK	Processor	
27:20	Data Byte Low – 8 bits	External Controller	Loaded into the Notify Data Low Byte Register
28	ACK	Processor	
36:29	Data Byte High – 8 bits	External Controller	Loaded into the Notify Data High Byte Register
37	ACK	Processor	
38	Stop	External Controller	

Format of Read Command

The external controller performs Byte Read commands to the Processor SMBus Target interface. The "Command" field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 119. Target Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Micro controller	
2–8	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
9	Write	External Micro controller	Always 0
10	ACK	Processor	
11–18	Command code – 8 bits	External Micro controller	Indicates which register is being accessed. Refer to the Tale below for a list of implemented registers.
19	ACK	Processor	
20	Repeated Start	External Micro controller	
21–27	Target Address - 7 bits	External Micro controller	Must match value in Receive Target Address register
28	Read	External Micro controller	Always 1
29	ACK	Processor	
30–37	Data Byte	Processor	Value depends on register being accessed. Refer to the Table below for a list of implemented registers.
38	NOT ACK	External Micro controller	
39	Stop	External Micro controller	

Table 120. Data Values for Target Read Registers

Register	Bits	Description
0	7:0	Reserved for capabilities indication. Should always return 00h. Future chips may return another value to indicate different capabilities.
1	2:0	**System Power State** 000 = S0 100 = S4 101 = S5 Others = Reserved

continued...

Register	Bits	Description
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** *Note:* The Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the Processor will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect.** 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	**Temperature Event.** 1 = Temperature Event occurred. This bit will be set if the Processor 's THRM# input signal is active. Else this bit will read "0."
	2	**DOA Processor Status**. This bit will be 1 to indicate that the processor is dead
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status:** Reflects the value of the GPIO11/SMBALERT# pin (when the pin is configured as SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always return 1 if SMBALERT_DISABLE = 1. (high = 1, low = 0).
5	0	**FWH bad bit:** This bit will be 1 to indicate that the FWH read returned FFh, which indicates that it is probably blank.
	1	**Battery Low Status:** 1 if the BATLOW# pin is a 0.
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD:** Indicates the failure core power well ramp during boot/resume. This bit will be active if the PLT_PWROK pin is not asserted.
	6	**Thermal Trip:** This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message.
	7	Reserved: Default value is "X" *Note:* Software should not expect a consistent value when this bit is read through SMBUS/SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC

continued...

Register	Bits	Description
E	7:0	Month of the RTC
F	7:0	Year of the RTC
10h–FFh	7:0	Reserved

Table 121. Enables for SMBus Target Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	Event
Target Write to Wake/SMI# Command	X	X	Wake generated when asleep. Target SMI# generated when awake (SMBUS_SMI_STS)
Target Write to SMLINK_SLAVE_SMI Command	X	X	Target SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

37.2 SMBus Power Gating

SMBus shares the Power Gating Domain with Primary-to-Sideband Bridge (P2SB). A single FET controls the single Power Gating Domain; but SMBus and P2SB each has its own dedicated Power Gating Control Block. The FET is only turned off when all these interfaces are ready to PG entry or already in the PG state.

37.3 Signal Description

Signal Name	Type	Description	Availability
GPP_C00/**SMBCLK**/ USB-C_GPP_C00	I/OD	**SMBus Clock:** External Pull-up resistor is required.	H/U-Series Processor
GPP_C01/ **SMBDATA**/USB-C_GPP_C01	I/OD	**SMBus Data:** External Pull-up resistor is required.	H/U-Series Processor
GPP_C02/ **SMBALERT#**/USB-C_GPP_C02	I/OD	**SMBus Alert:** This signal is used to wake the system or generate SMI#. External Pull-up resistor is required.	H/U-Series Processor

37.4 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SMBALERT#	Pull-down	20 kohm ± 30%	The internal pull-down resistor is disable after RSMRST# de-asserted.

37.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
SMBDATA	Primary	Undriven	Undriven	Undriven
SMBCLK	Primary	Undriven	Undriven	Undriven
SMBALERT#	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

38.0 Serial Peripheral Interface (SPI)

The processor provides Serial Peripheral Interfaces (SPI) to connect up to two flash devices. The SPI0 interface consists of three Chip Select signals. SPI0 interface can allow two flash memory devices (SPI0_CS0# and SPI0_CS1#) and one TPM device (SPI0_CS2#) to be connected to the processor. The SPI0 interface supports 1.8 V only.

Table 122. Acronyms

Acronyms	Description
CLK	Clock
CS	Chip Select
FCBA	Flash Component Base Address
FLA	Flash Linear Address
FMBA	Flash Controller Base Address
FPSBA	Flash Processor Strap Base Address
FRBA	Flash Region Base Address
MDTBA	MIP Descriptor Table Base Address
MISO	Terminology to indicate signal direction: input to the host, output from the device
MOSI	Terminology to indicate signal direction: output from the host, input to the device
TPM	Trusted Platform Module

38.1 Functional Description

38.1.1 SPI0 for Flash

The Serial Peripheral Interface (SPI0) supports two SPI flash devices via two chip select (SPI0_CS0# and SPI0_CS1#). The maximum size of flash supported is determined by the SFDP-discovered addressing capability of each device. Each component can be up to 16 MB (32 MB total addressable) using 3-byte addressing. Each component can be up to 64 MB (128 MB total addressable) using 4-byte addressing. Another chip select (SPI0_CS2#) is also available and only used for TPM on SPI support. The processor drives the SPI0 interface clock at either 14 MHz, 25 MHz, 33 MHz and 50 MHz and will function with SPI flash/TPM devices that support at least one of these frequencies. The SPI interface supports 1.8 V only

A SPI0 flash device supporting SFDP (Serial Flash Discovery Parameter) is required for all design. A SPI0 flash device on SPI0_CS0# with a valid descriptor must be attached directly to the processor.

The processor supports fast read which consist of:

1. Dual Output Fast Read (Single Input Dual Output)

2. Dual I/O Fast Read (Dual Input Dual Output)
3. Quad Output Fast Read (Single Input Quad Output)
4. Quad I/O Fast Read (Quad Input Quad Output)

The processor SPI0 has a third chip select SPI0_CS2# for TPM support over SPI. The TPM on SPI0 will use SPI0_CLK, SPI0_MISO, SPI0_MOSI and SPI0_CS2# SPI signals.

SPI0 Supported Features

- **Descriptor Mode**

  Descriptor Mode is required for all SKUs of the processor. Non-Descriptor Mode is not supported.

- **SPI0 Flash Regions**

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

Table 123. SPI0 Flash Regions

Region	Content
0	Flash Descriptor
1	BIOS
2	Intel ®CSME
3	GbE - Location for Integrated LAN firmware and MAC address
4	PDR - Platform Data Region
8	EC - Embedded Controller
10	Intel® Silicon Security Engine

Only four controllers can access the regions: Host processor running BIOS code, Integrated Gigabit Ethernet and Host processor running Gigabit Ethernet Software, Intel Converged Security and Management Engine, and the EC.

The Flash Descriptor and Intel® CSME region are the only required regions. The Flash Descriptor has to be in region 0 and region 0 must be located in the first sector of Device 0 (Offset 0). All other regions can be organized in any order.

Regions can extend across multiple components, but must be contiguous.

Flash Region Sizes

SPI0 flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4 KB or larger block. GbE requires two 4 KB or larger blocks. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the Intel® CSME and BIOS regions. The Intel® CSME region contains firmware to support Intel Active Management Technology and other Intel® CSME capabilities.

Table 124. 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

continued...

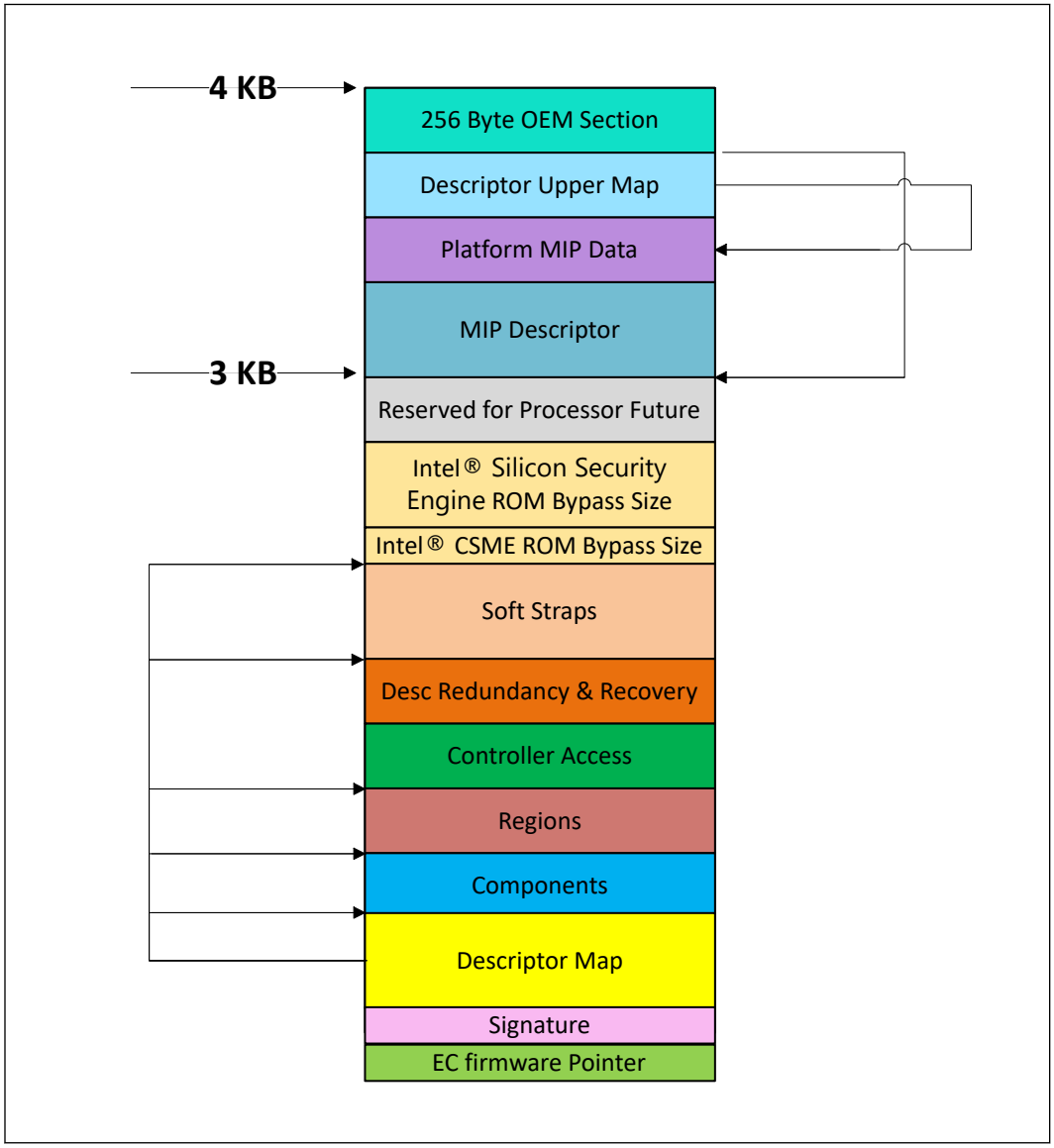
Region	Size with 4 KB Blocks	Size with 8 KB Blocks	Size with 64 KB Blocks
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSME	Varies by Platform	Varies by Platform	Varies by Platform
EC	Varies by Platform	Varies by Platform	Varies by Platform
PDR	Varies by Platform	Varies by Platform	Varies by Platform
Intel® CSME Data	Varies by Platform	Varies by Platform	Varies by Platform

Flash Descriptor

The bottom sector of the flash component 0 contains the Flash Descriptor. The maximum size of the Flash Descriptor is 4 KB. If the block/sector size of the SPI0 flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. It requires its own discrete erase block, so it may need greater than 4 KB of flash space depending on the flash architecture that is on the target system. Two additional redundant back-ups of the Flash Descriptor have been added for data resilience.The information stored in the Flash Descriptor can only be written during the manufacturing process as its read/write permissions must be set to read only when the computer leaves the manufacturing floor.

The Flash Descriptor is made up of fifteen sections as shown in the figure below:

Figure 31. Flash Descriptor Regions



- EC Firmware Pointer is located in the first 16 bytes of the Descriptor and contains the address location for EC flash region. The format for the EC Firmware Pointer address is dependent on EC vendors/OEM implementation of this field.
- The Flash signature at the bottom of the flash (offset 10h) must be 0FF0A55Ah in order to be in Descriptor mode.
- The Descriptor map has pointers to the lower five descriptor sections as well as the size of each.
- The Component section has information about the SPI flash part(s) the system. It includes the number of components, density of each component, read, write and erase frequencies and invalid instructions.

- The Region section defines the base and the limit of the BIOS, IFWI, GbE, Platform Data Region (PDR- Optional), Embedded Controller (EC- Optional) regions as well as their size.
- The processor soft strap sections contain configurable parameters.
- The Reserved region is for future processor usage.
- The Descriptor Upper Map determines the length and base address of the Intel ® CSME VSCC Table.
- The Intel ® CSME VSCC Table holds the JEDEC ID and the ME VSCC information for all the SPI Flash part(s) supported by the NVM image. BIOS and GbE write and erase capabilities depend on VSCC0 and VSCC1 registers in SPIBAR memory space.
- OEM Section is 256 bytes reserved at the top of the Flash Descriptor for use by OEM.

Descriptor Controller Region

The Controller region defines read and write access setting for each region of the SPI0 device. The Controller region recognizes four Controllers: BIOS, Gigabit Ethernet, Intel® CSME, and EC. Each Controller is only allowed to do direct reads of its primary regions.

Table 125. Region Access Control Table

Controller Read/Write Access				
Region	**Processor and BIOS**	**Intel® CSME**	**GbE Controller**	**EC**
Descriptor (0)	Read Only	Read Only	Not Accessible	Not Accessible
BIOS (1)	processor / BIOS can always read from and write to BIOS region prior to EOP	Not Accessible	Not Accessible	Not Accessible
Intel® CSME (2)	Read/Write (BIOS Only)	Intel® CSME can always read from and write to firmware region	Not Accessible	Not Accessible
Gigabit Ethernet (3)	Not Accessible	Read/Write	GbE software can always read from and write to GbE region	Not Accessible
PDR (4)	Not Accessible	Not Accessible	Not Accessible	Not Accessible
EC (8)	Read/Write	Not Accessible	Not Accessible	EC can always read from and write to EC region.
Intel® CSME Data (15)	Not Accessible	Read/Write	Not Accessible	Not Accessible

Notes:
- The Region Access values listed above represent post manufacturing configuration only.
- Descriptor and PDR region is not a Controller, so they will not have Controller R/W access.
- Descriptor should NOT have write access by any Controller in production systems.
- PDR region should only have read and/or write access by processor/Host. GbE and Intel® CSME should NOT have access to PDR region.

Table 126. Flash Descriptor Processor Complex Soft Strap

Region Name	Starting Address
Signature	10h
Component FCBA	30h
Regions FRBA	40h
Controllers FMBA	80h
Desc Redundancy & Recovery	320h
MDTBA	C00h
IOE PMC Straps	C6Ch
IOE Soft Straps	CACh
Processor Straps	CECh D8Ch
Intel® CSME Straps	D9Ch

Flash Access

There are two types of accesses: Direct Access and Program Register Accesses.

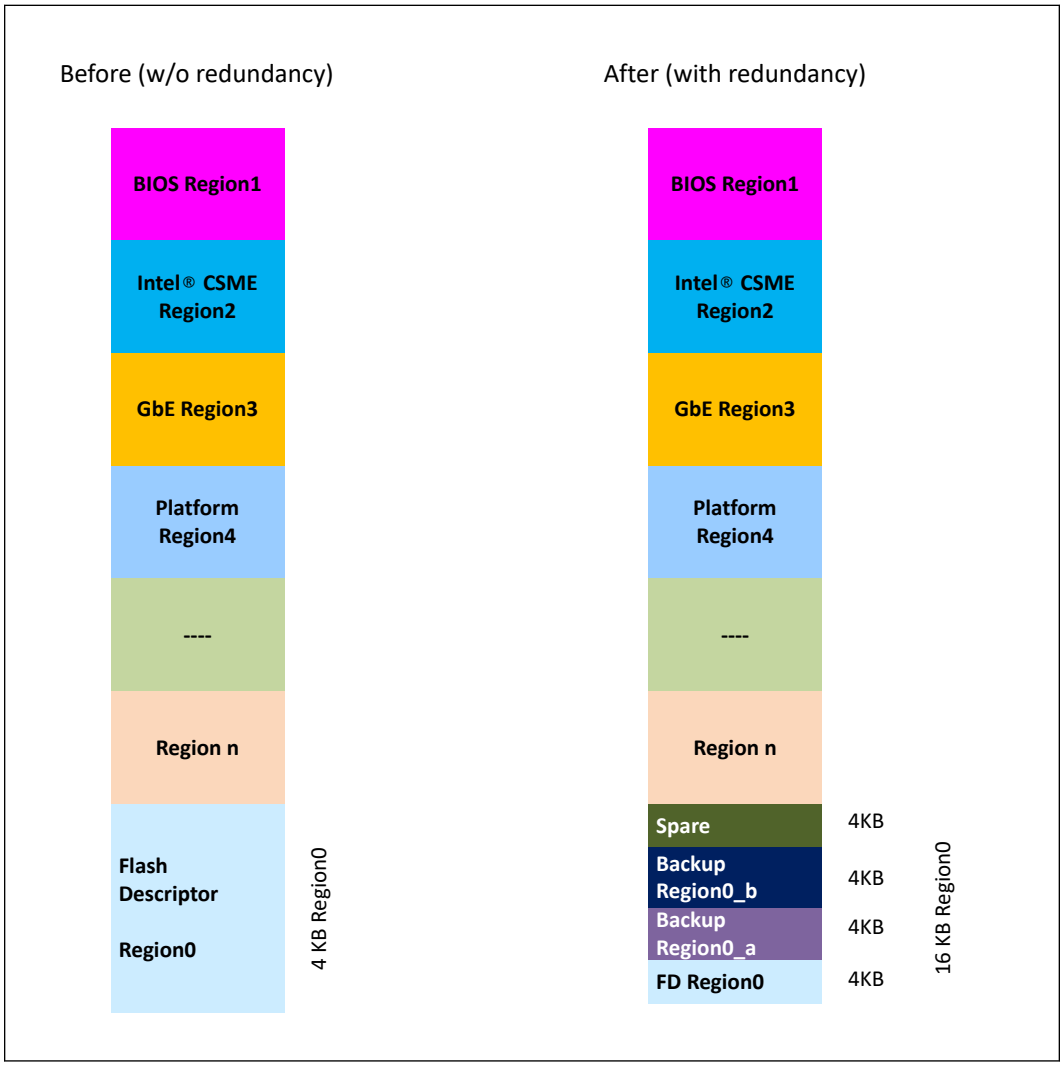
- **Direct Access**
  - Controllers are allowed to do direct read only of their primary region
    - Gigabit Ethernet region can only be directly accessed by the Gigabit Ethernet controller. Gigabit Ethernet software must use Program Registers to access the Gigabit Ethernet region.
  - Controller's Host or Management Engine virtual read address is converted into the SPI0 Flash Linear Address (FLA) using the Flash Descriptor Region Base/ Limit registers

    Direct Access Security
  - Requester ID of the device must match that of the primary Requester ID in the Controller Section
  - Calculated Flash Linear Address must fall between primary region base/limit
  - Direct Write not allowed
  - Direct Read Cache contents are reset to 0's on a read from a different Controller
- **Program Register Access**
  - Program Register Accesses are not allowed to cross a 4 KB boundary and cannot issue a command that might extend across two components
  - Software programs the FLA corresponding to the region desired
    - Software must read the devices Primary Region Base/Limit address to create a FLA.
  - *Register Access Security*
    Only primary region Controllers can access the registers

Flash Descriptor Redundancy and Recovery

In order to provide descriptor redundancy and recovery, SPI flash controller uses two 4 KB spaces or regions as the backup descriptor regions. Each backup descriptor region size is 4 KB.

Figure 32. Flash Descriptor Redundancy



In the main and backup descriptor regions, the following fields are defined for the descriptor integrity check and recovery. Before SPI controller reads the descriptor, it

- Reads Main Descriptor Region and calculates SHA-256 hash.
- Reads Active Backup Descriptor Region and calculates hash.
- Compares each hash result with the hash in that region.
- Takes action based on result and policy byte (in Main Descriptor).

RPMC Configuration

Intel Replay Protection Monotonic Counter (RPMC) is a capability providing Anti-Replay Protection using Monotonic Counters inside SPI Flash. Intel RPMC is a critical security feature designed to protect the SPI part of Intel platforms from unauthorized write operations. This innovative technology acts as a robust defense mechanism, ensuring that only authorized write operations are permitted, thus preventing any unauthorized access to the SPI.

RPMC protection relies on:

- Special RPMC HW and logic inside the SPI Flash.
- Intel CSME FW support that utilizes RPMC capabilities within Flash.

At the core of RPMC's functionality lies the concept of the session key.

The session key is a cryptographic key derived from several factors residing on the processor . These factors are carefully selected and stored upon provisioning RPMC to the SPI part. The session key serves as a means of authenticating each incoming write message to the SPI. When an authorized operation is initiated, the session key is used to verify the legitimacy of the request. If the session key does not match the expected value, the SPI part will reject the request, effectively blocking malicious or unauthorized write operations.

Furthermore, the session key also extends its protective shield to cover a specific set of sensitive read messages. This holistic approach ensures that not only write operations but also read operations involving sensitive data are monitored and authenticated, enhancing the overall security of the system.

Two features of RPMC can be enabled:

- RPMC will be enabled on platforms with RPMC SPI. During Intel End of Manufacturing the processor will be bound with RPMC SPI
- When SPI is replaced, re-binding between the new RPMC SPI and theprocessor will happen automatically on first boot.

Monotonic Counters

Monotonic counters are counters on the SPI Flash maintained by Intel CSME FW. SPI Flash has a set of four 32-bit monotonic counters, where Intel CSME FW uses two of these counters. Intel CSME FW ensures FW write operations will not exceed SPI RPMC monotonic counter increment rate specified by RPMC HW during platform lifetime supported by Intel. Reading and incrementing the counters in the Flash is done using authenticated commands with a key known to both: SPI Flash and Intel® CSME FW

Binding at End of Manufacturing (EOM)

RPMC Binding pairs between SPI Flash and the processor by provisioning the Binding key produced by the processor into SPI Flash. This pairing is done as part of the EOM flow which usually takes place at the manufacturing line.

In conclusion, Intel RPMC, with its Replay Monotonic Counter and session key mechanism, stands as a powerful safeguard against unauthorized write operations and unauthorized access to sensitive data in the SPI part. This robust security feature, derived from the session key, adds an additional layer of protection to Intel platforms, making them more resilient against potential threats and ensuring the integrity and confidentiality of the data stored in the SPI.

38.1.2 SPI0 Support for TPM

The processorSPI0 flash controller supports a discrete TPM on the platform via its dedicated SPI0_CS2# signal. The platform must have no more than 1 TPM.

SPI0 controller supports accesses to SPI0 TPM at approximately 17 MHz, 33 MHz and 48 MHz depending on the soft strap. 20 MHz is the reset default, a valid 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 processor interrupt configurable GPIO pin. Thus, TPM interrupt is completely independent from the SPI0 controller.

38.2 Signal Description

Signal Name	Type	Description
SPI0_CLK	O	**SPI0 Clock**: SPI clock signal for the common flash/TPM interface. Supports upto maximum of 50 MHz.
SPI0_CS0#	O	**SPI0 Chip Select 0**: Used to select the primary SPI0 Flash device. *Note:* This signal cannot be used for any other type of device than SPI Flash.
SPI0_CS1#	O	**SPI0 Chip Select 1**: Used to select an optional secondary SPI0 Flash device. *Note:* This signal cannot be used for any other type of device than SPI Flash.
SPI0_CS2#	O	**SPI0 Chip Select 2**: Used to select the TPM device if it is connected to the SPI0 interface. It cannot be used for any other type of device.
SPI0_MOSI	I/O	**SPI0 Host OUT Device IN**: Defaults as a data output pin for the processor in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI0_IO0) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_MISO	I/O	**SPI0 Host IN Device OUT**: Defaults as a data input pin for the processor in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI0_IO1) to support the Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_IO2	I/O	**SPI0 Data I/O**: A bidirectional signal used to support Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes. This signal is not used in Dual Output Fast Read mode.
SPI0_IO3	I/O	**SPI0 Data I/O**: A bidirectional signal used to support Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes. This signal is not used in Dual Output Fast Read mode.

38.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SPI0_CLK	Pull-down	20 kohm ± 30%	
SPI0_MOSI	Pull-up	20 kohm ± 30%	Note
SPI0_MISO	Pull-up	20 kohm ± 30%	Note
SPI0_CS[2:0]#	Pull-down	20 kohm ± 30%	
SPI0_IO[2:3]	Pull-up	20 kohm ± 30%	

NOTE

Above resistor type is dynamic state controlled by the SPI controller. The internal Pull-up is disabled when RSMRST# is asserted (during reset) and only enabled after RSMRST# de-assertion.

38.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
SPI0_CLK	Primary	Internal Pull-down	Driven Low	Driven Low
SPI0_MOSI	Primary	Hi-Z	Internal Pull-up , then Driven Low	Driven Low
SPI0_MISO	Primary	Hi-Z	Internal Pull-up	Internal Pull-up
SPI0_CS[2:0]#	Primary	Internal Pull-down	Driven High	Driven High
SPI0_IO[3:2]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up
Note: 1. During reset refers to when RSMRST# is asserted.				

39.0 Enhanced Serial Peripheral Interface (eSPI)

The processor provides the Enhanced Serial Peripheral Interface (eSPI) to support connection of an EC (typically used in mobile platform) or an SIO (typically used in desktop platform) to the platform. Below are the key features of the interface:

- 1.8 V support only
- Support for Host Attached Flash (MAF) and Device Attached Flash (SAF).
- Support for up to 50 MHz (configured by soft straps)
- Up to quad mode support
- Support for PECI over eSPI
- Support for Multiple OOB Controller (dedicated OOB channel for different OOB Controllers in the Processor such as PMC and CSME)
- Transmitting RTC time/date to the device upon request
- In-band messages for communication between the Processor and device to eliminate side-band signals.
- Real time SPI flash sharing, allowing real time operational access by the processor and device.

Table 127. Acronyms

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

Table 128. References

Specification	Document Number/Location
Enhanced Serial Peripheral Interface (eSPI) Specifications	https://downloadcenter.intel.com/download/27055/eSPI

39.1 Functional Description

39.1.1 Operating Frequency

The eSPI controller supports 20 MHz, 25 MHz, 33 MHz, and 50 MHz. A eSPI device can support frequencies lower than the recommended maximum frequency (50 MHz). In addition, the eSPI device must support a minimum frequency of 20 MHz for default (reset) communication between the host and device.

39.1.2 WAIT States from eSPI device

There are situations when the device cannot predict the length of the command packet from the controller. For non-posted transactions, the device 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 device after the TAR cycles.

39.1.3 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 device in a response to a Set_Configuration command), the processor 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 device.

39.1.4 Flash Sharing Mode

eSPI supports both Host and Device Attached Flash sharing (abbreviated in this as MAFS and SAFS, respectively).

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

39.1.5 PECI Over eSPI

When PECI Over eSPI is enabled, the eSPI device (i.e. EC) can access the processor PECI interface via eSPI controller, instead of the physical PECI pin. The support can improve the PECI responsiveness, and reduce PECI pins.

The PECI bus may be connected to the Processor via either the legacy PECI pin or the eSPI interface. Either of the operation via legacy PECI pin or over eSPI is enabled at a time in a given platform.

PECI over eSPI is not supported in Sx state. EC/BMC is not allowed to send the PECI command to eSPI in Sx states. More specifically, EC can only send PECI requests after VW PLT_RST# de-assertion.

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

39.1.6 Multiple OOB processes

The processor typically has multiple technology (Intel ®CSME, PMC, ISH, etc.). From an eSPI perspective, these are all classified as Out-of-Band (OOB) processes (as distinct from the Host). Since any of these OOB processes may need to communicate with the embedded controller on the platform (example, EC/BMC), the eSPI controller implements dedicated OOB channel for each OOB processes including PMC and Intel® CSME to improve the interface performance and potentially enable new usage models.

39.1.7 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 device is discovered through the GET_CONFIGURATION command issued by the processor to the eSPI device during initialization.

Table below summarizes the eSPI channels and supported transactions.

Table 129. eSPI Channels and Supported Transactions

CH #	Channel	Posted Cycles Supported	Non-Posted Cycles Supported
0	Peripheral	Memory Write, Completions	Memory Read, I/O Read/Write
1	Virtual Wire	Virtual Wire GET/PUT	N/A
2	Out-of-Band Message	SMBus Packet GET/PUT	N/A
3	Flash Access	N/A	Flash Read, Write, Erase
N/A	General	Register Accesses	N/A

Peripheral Channel (Channel 0) Overview

The Peripheral channel performs the following functions:

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

Virtual Wire Channel (Channel 1) Overview

The Virtual Wire channel uses a standard message format to communicate several types of signals between the components on the platform.

- **Sideband and GPIO Pins**: System events and other dedicated signals between the processor and eSPI device. These signals are tunneled between the 2 components over eSPI.
- **Serial IRQ Interrupts**: Interrupts are tunneled from the eSPI device to the processor. Both edge and triggered interrupts are supported.
- **eSPI Virtual Wires (VW)**

  Table below summarizes the virtual wires in eSPI mode.

Table 130. eSPI Virtual Wires (VW)

Virtual Wire	processor Pin Direction	Reset Control	Pin Retained in processor (For Use by Other Components)
SUS_STAT#	Output	ESPI_RESET#	No
PRIM_PWRDN_ACK	Output	ESPI_RESET#	No

continued...

Virtual Wire	processor Pin Direction	Reset Control	Pin Retained in processor (For Use by Other Components)
SUSWARN#	Output	ESPI_RESET#	No
SUS_ACK	Input	ESPI_RESET#	No
PLTRST#	Output	ESPI_RESET#	Yes
PME# (eSPI Peripheral PME)	Input	ESPI_RESET#	N/A
WAKE#	Input	ESPI_RESET#	No
SMI#	Input	PLTRST#	N/A
SCI#	Input	PLTRST#	N/A
RCIN#	Input	PLTRST#	No
SLP_A#	Output	ESPI_RESET#	Yes
SLP_S3#/SLP_S4#/SLP_S5#/ SLP_LAN#/SLP_WLAN#	Output	RSMRST#	Yes
DEVICE_BOOT_LOAD_DONE	Input	ESPI_RESET#	N/A
DEVICE_BOOT_LOAD_STATUS	Input	ESPI_RESET#	N/A
HOST_RST_WARN	Output	PLTRST#	N/A
HOST_RST_ACK	Input	PLTRST#	N/A
OOB_RST_WARN	Output	ESPI_RESET#	N/A
OOB_RST_ACK	Input	ESPI_RESET#	N/A
HOST_C10	Output	PLTRST#	N/A
ERROR_NONFATAL	Input	ESPI_RESET#	N/A
ERROR_FATAL	Input	ESPI_RESET#	N/A
DNX_WARN	Output	PLTRST#	N/A
DNX_ACK	Input	ESPI_RESET#	N/A

- **Interrupt Events**

  eSPI supports both level and edge-triggered interrupts. Refer to the eSPI Specification for details on the theory of operation for interrupts over eSPI.

  The eSPI controller will issue a message to the interrupt controller when it receives an IRQ group in its VW packet, indicating a state change for that IRQ line number.

  The eSPI device can send multiple VW IRQ index groups in a single eSPI packet, up to the Operating Maximum VW Count programmed in its Virtual Wire Capabilities and Configuration Channel.

  The eSPI controller acts only as a transport for all interrupt events generated from the device. It does not maintain interrupt state, polarity or enable for any of the interrupt events.

Out-of-Band Channel (Channel 2) Overview

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

- **Tunnel MCTP Packets between the Intel® CSME and eSPI Device**: The Intel® CSME communicates MCTP messages to/from the device by embedding those packets over the eSPI protocol. This eliminates the SMBus connection between the processor and the device which was used to communicate the MCTP messages. The eSPI controller simply acts as a message transport and forwards the packets between the Intel® CSME and eSPI device.
- **Tunnel Processor Temperature Data to the eSPI device**: The eSPI controller stores the processor temperature data internally and sends it to the device using a posted OOB message when a request is made to a specific destination address.
- **Tunnel Processor RTC Time and Date Bytes to the eSPI device**: the eSPI controller captures this data internally at periodic intervals from the processor RTC controller and sends it to the device using a posted OOB message when a request is made to a specific destination address.
- **Processor Temperature Data Over eSPI OOB Channel**

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

  The packet formats for the temperature request from the eSPI device and the processor response back are shown in the two figures below.

Figure 33. eSPI Device Request to Processor for Processor Temperature

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 04h							
3	Destination Device Addr. = 01h (OOB HW Handler)							0
4	Common code = 01h (Get_Temp)							
5	Byte Count = 01h							
6	Source Device Address[7:0] = 0Fh (eSPI Device 0/EC)							1

Figure 34. Processor Response to eSPI device with Processor Temperature

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 05h							
3	Destination Device Addr. = 0Eh (eSPI Device 0/EC)							0
4	Common code = 01h (Get_Temp)							
5	Byte Count = 02h							
6	Source Device Address [7:0] = 01h (OOB HW Handler)							1
7	Temperature Data [7:0]							

- **Processor RTC Time/Date to EC Over eSPI OOB Channel**

The processor eSPI controller supports the transmitting of processor RTC time/date to the eSPI device. This allows the eSPI device to synchronize with the Processor RTC system time. Moreover, using the OOB message channel allows reading of the internal time when the system is in Sx states.

The RTC time consists of 7 bytes: seconds, minutes, hours, day of week, day of month, month and year. The controller provides all the time/date bytes together in a single OOB message packet. This avoids the boundary condition of possible roll over on the RTC time bytes if each of the hours, minutes, and seconds bytes is read separately.

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

Figure 35. eSPI Device Request to Processor for Processor RTC Time

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 04h							
3	Dest Device Addr. [7:1] = 01h (OOB HW Handler)							0
4	Common code = 02h (Get_RTC_Time)							
5	Byte Count = 01h							
6	Source Device Address [7:0] = 0Fh (eSPI Device 0/EC)							1

Figure 36. Processor Response to eSPI device with RTC Time

Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0]= 0Ch							
3	Dest Device Addr. [7:0] = 0Eh (eSPI Device 0/EC)							0
4	Common code = 02h (Get_RTC_Time)							
5	Byte Count = 09h							
6	Source Device Address [7:1] = 01h (OOB HW Handler)							1
7	Reserved					DM	HF	DS
8	RTC Time: Seconds							
9	RTC Time: Minutes							
10	RTC Time: Hours							
11	RTC Time: Day of Week							
12	RTC Time: Day of Month							
13	RTC Time: Month							
14	RTC Time: Year							

NOTES

1. **DS**: Daylight Savings. A 1 indicates that Daylight Saving has been comprehended in the RTC time bytes. A 0 indicates that the RTC time bytes do not comprehend the Daylight Savings.
2. **HF**: Hour Format. A 1 indicates that the Hours byte is in the 24-hr format. A 0 indicates that the Hours byte is in the 12-hr format. In 12-hr format, the seventh bit represents AM when it is a 0 and PM when it is a 1.
3. **DM**: Data Mode. A 1 indicates that the time byte are specified in binary. A 0 indicates that the time bytes are in the Binary Coded Decimal (BCD) format.

Flash Access Channel (Channel 3) Overview

The Flash Access channel supports the Host Attached Flash (MAF) configuration, where the flash device is directly attached to the processor. This configuration allows the eSPI device to access the flash device attached to the processor through a set of flash access commands. These commands are routed to the flash controller and the return data is sent back to the eSPI device.

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

- **Host Attached Flash Channel Controller (MAFCC) Flash Operations and Addressing**

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

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

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

- **Device Attached Flash Channel Controller (SAFCC) Flash Operation and Addressing**

  The processor is allocated dedicated regions (for each of the supported Controllers) within the eSPI SAFCC. The processor has read, write, and erase access to these regions, as well as any other regions that maybe permitted by the region protections set in the Flash Descriptor.

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

  The SAFCC supports Flash Read, Write and Erase operations. It also supports Read SFDP and Read JEDEC ID commands as specified in the eSPI Specification for Server platforms.

39.2 Signal Description

Signal Name	Type	Description	Availability
GPP_A00/**ESPI_IO0**/USB-C_GPP_A00	I/O	**eSPI Data Signal 0:** Bi-directional pin used to transfer data between the Processor and eSPI device.	H/U-Series Processor
GPP_A01/**ESPI_IO1**/USB-C_GPP_A01	I/O	**eSPI Data Signal 1:** Bi-directional pin used to transfer data between the Processor and eSPI device	H/U-Series Processor
GPP_A02/**ESPI_IO2**/ PRIMPWRDNACK/USB-C_GPP_A02	I/O	**eSPI Data Signal 2:** Bi-directional pin used to transfer data between the Processor and eSPI device	H/U-Series Processor
GPP_A03/**ESPI_IO3**/ PRIMACK#/USB-C_GPP_A03	I/O	**eSPI Data Signal 3:** Bi-directional pin used to transfer data between the Processor and eSPI device	H/U-Series Processor
GPP_A04/**ESPI_CS0#**/USB-C_GPP_A04	O	**eSPI Chip Select 0:** Driving CS# signal low to select eSPI device for the transaction.	H/U-Series Processor
GPP_A05/**ESPI_CLK**/USB-C_GPP_A05	O	**eSPI Clock:** eSPI clock output from the Processor to device.	H/U-Series Processor
GPP_A06/**ESPI_RESET#**/ USB-C_GPP_A06	O	**eSPI Reset:** Reset signal from the Processor to eSPI device.	H/U-Series Processor

39.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes

39.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ESPI_IO[3:0]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up
ESPI_CLK	Primary	Internal Pull-down	Driven Low	Driven Low

continued...

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
ESPI_ CS0#	Primary	Internal Pull-up	Driven High	Driven High
ESPI_RESET#	Primary	Driven Low	Driven High	Driven High
Note: Reset reference for primary well pins is RSMRST#.				

40.0 Intel® Serial IO Generic SPI (GSPI) Controllers

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

Each interface consists of a clock (CLK), one chip selects (CS) and two data lines (MOSI and MISO).

The GSPI interfaces support the following features:

- 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)
- Full duplex synchronous serial interface
- Supports the Motorola's* SPI protocol
- Operates in Host mode only

NOTE

Device mode is not supported.

Table 131. Acronyms

Acronyms	Description
GSPI	Generic Serial Peripheral Interface
LTR	Latency Tolerance Reporting

40.1 Functional Description

40.1.1 Controller Overview

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

The processor or DMA accesses data through the GSPI ports transmit and receive FIFOs.

A processor access takes the form of programmed I/O, transferring one FIFO entry per access. Processor accesses must always be 32 bits wide. Processor writes to the FIFOs are 32 bits wide, but the Processor will ignore all bits beyond the programmed FIFO data size. Processor reads to the FIFOs are also 32 bits wide, but the receive data written into the Receive FIFO is stored with '0' in the most significant bits (MSB) down to the programmed data size.

The FIFOs can also be accessed by DMA, which must be in multiples of 1, 2, or 4 bytes, depending upon the EDSS value, and must also transfer one FIFO entry per access.

For writes, the Enhanced SPI takes the data from the transmit FIFO, serializes it, and sends it over the serial wire to the external peripheral. Receive data from the external peripheral on the serial wire is converted to parallel words and stored in the receive FIFO.

A programmable FIFO trigger threshold, when exceeded, generates an interrupt or DMA service request that, if enabled, signals the processor or DMA respectively to empty the Receive FIFO or to refill the Transmit FIFO.

The GSPI controller, as a host, provides the clock signal and controls the chip select line. Commands codes as well as data values are serially transferred on the data signals. The Processor asserts a chip select line to select the corresponding peripheral device with which it wants to communicate. The clock line is brought to the device whether it is selected or not. The clock serves as synchronization of the data communication.

40.1.2 DMA Controller

The GSPI controllers have an integrated DMA controller.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires that the peripheral control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires that the peripheral control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory. The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode.

Channel Control

- The source transfer width and destination transfer width are programmable. The width can be programmed to 1, 2, or 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. this number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual Channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. the block size is not limited by the source or destination transfer widths.

- Address incrementing modes: The DMA has a configurable mechanism for computing the source and destination addresses for the next transfer within the current block. The DMA supports incrementing addresses and constant addresses.
- Flexibility to configure any hardware handshake sideband interface to any of the DMA channels.
- Early termination of a transfer on a particular channel.

40.1.3 Reset

Each host controller has an independent rest associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered ON and require SW (BIOS or driver) to write into the corresponding reset register to bring the controller from reset state into operational mode.

40.1.4 Power Management

Device Power Down Support

In order to power down peripherals connected to the Processor GSPI bus, the idle configured state of the I/O signals must be retained to avoid transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when the bus is powered off (power gated). The Processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. However, the GSPI bus architecture does not provide the architectural means to define dynamic latency tolerance messaging. Therefore, the interface supports this by reporting its service latency requirements to the platform power management controller via LTR registers.

The controller's latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements. In this scheme, the latency requirement is a function of the controller state. The latency for transmitting data to/from its connected device at a given rate while the controller is active is representative of the active latency requirements. On the other hand if the device is not transmitting or receiving data and idle, there is no expectation for end to end latency.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end-to-end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

40.1.5 Interrupts

Each interface has the ability to interrupt and notify the driver that service is required

When an interrupt occurs, the device driver needs to read both the host controller and DMA interrupt status and transmit completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

40.1.6 Error Handling

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

40.2 Signal Description

Signal Name	Type	Description
GPP_E10/THC0_SPI1_CS#/ **GSPI0_CS0#**/USB-C_GPP_E10	O	**Generic SPI 0 Chip Select**
GPP_E11/THC0_SPI1_CLK/ **GSPI0_CLK**/USB-C_GPP_E11	O	**Generic SPI 0 Clock**
GPP_E12/THC0_SPI1_IO1/ **GSPI0_MISO**/I2C4_SDA/USB-C_GPP_E12	I	**Generic SPI 0 MISO**
GPP_E13/THC0_SPI1_IO0/ **GSPI0_MOSI**/I2C4_SCL/USB-C_GPP_E13	O	**Generic SPI 0 MOSI**
GPP_F17/THC1_SPI2_CS#/ ISH_SPIA_CS#/**GSPI1_CS0#**/ USB-C_GPP_F17	O	**Generic SPI 1 Chip Select 0**
GPP_F11/THC1_SPI2_CLK/ ISH_SPIA_CLK/**GSPI1_CLK**/ USB-C_GPP_F11	O	**Generic SPI 1 Clock**
GPP_F13/GSXSLOAD/ THC1_SPI2_IO1/ ISH_SPIA_MOSI/**GSPI1_MISO**/ I2C5_SDA/USB-C_GPP_F13	I	**Generic SPI 1 MISO**
GPP_F12/GSXDOUT/ THC1_SPI2_IO0/ ISH_SPIA_MISO/**GSPI1_MOSI**/ I2C5_SCL/USB-C_GPP_F12	O	**Generic SPI 1 MOSI**
GPP_F18/THC1_SPI2_INT#/ **GSPI0A_CS0#**/USB-C_GPP_F18	O	**Generic SPI 0A Chip Select**

continued...

Signal Name	Type	Description
GPP_F16/GSXCLK/ THC1_SPI2_RST#/ **GSPI0A_CLK**/USB-C_GPP_F16	O	**Generic SPI 0A Clock**
GPP_F15/GSXSRESET#/USB-C_SMLDATA/THC1_SPI2_IO3/ **GSPI0A_MISO**/USB-C_GPP_F15	I	**Generic SPI 0A MISO**
GPP_F14/GSXDIN/USB-C_SMLCLK/THC1_SPI2_IO2/ **GSPI0A_MOSI**/USB-C_GPP_F14	O	**Generic SPI 0A MOSI**

40.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
GSPI0_MOSI	Pull Down	20 kohm± 30%	The integrated pull down is disabled after PLT_PWROK assertion
GSPI1_MOSI	Pull Down	20 kohm ± 30%	The integrated pull down is disabled after PLT_PWROK assertion
GSPI0_MISO	Pull Down	20 kohm ± 30%	
GSPI1_MISO	Pull Down	20 kohm ± 30%	

40.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
GSPI1_CS0#, GSPI0_CS0# , **GSPI0A_CS0#**	Primary	Undriven	Undriven	Undriven
GSPI1_CLK, GSPI0_CLK , **GSPI0A_CLK**	Primary	Undriven	Undriven	Undriven
GSPI1_MISO, GSPI0_MISO , **GSPI0A_MISO**	Primary	Undriven	Undriven	Undriven
GSPI1_MOSI, GSPI0_MOSI, **GSPI0A_MOSI**	Primary	Internal Pull-down	Driven Low	Internal Pull-down
Note: 1. Reset reference for primary well pins is RSMRST#.				

41.0 Touch Host Controller (THC)

Touch Host Controller provides a standard SPI interface for Processor to connect to external touch ICs. Only SPI IOs are supported.

THC also supports the GPIO based SPI interrupt from touch IC and supports hardware autonomous power management scheme within the Processor

Table 132. Acronyms

Acronyms	Description
CLK	Clock
CS	Chip Select
THC	Touch Host Controller

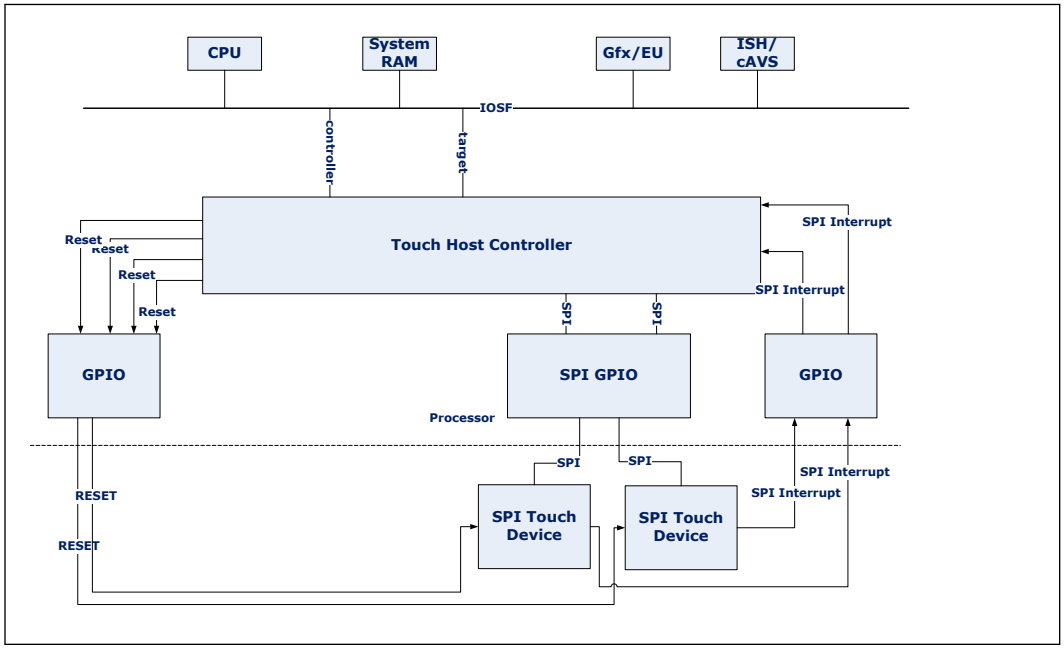
41.1 Functional Description

The Touch Host Controller (THC) supports a host controller interface to the touch IC for high bandwidth touch data transfer from SPI based touch ICs. THC provides high bandwidth DMA services to the touch driver and transfer the touch raw data or HID reports to internal touch accelerator (For example, graphics EUs or host Processor), or host driver respectively.

The THC controller bridges the Processor bus and SPI ports, below are the details.

- THC Controller
  - Touch Host controller bridges the Processor bus and SPI
  - The THC Controller has the following interfaces
- SPI IO
  - 1.8 V SPI IOs.
  - Provides SPI interface to the THC core.
  - Maximum Frequency supported 41.67 MHz.

Figure 37. THC Block Diagram



41.2 Signal Description

Signal Name	Type	Description
GPP_E11/**THC0_SPI1_CLK**/GSPI0_CLK/ USB-C_GPP_E11	O	**THC0_SPI1 Clock**: THC SPI1 clock output from Processor. Supports 42.67MHz.
GPP_F11/**THC1_SPI2_CLK**/ ISH_SPIA_CLK/GSPI1_CLK/USB-C_GPP_F11	O	**THC1_SPI2 Clock**: THC SPI2 clock output from Processor. Supports 42.67MHz.
GPP_E10/**THC0_SPI1_CS#**/ GSPI0_CS0#/USB-C_GPP_E10	O	**THC0_SPI1 Chip Select**: Used to select the touch devices if it is connected to THC0_SPI1 interface.
GPP_F17/**THC1_SPI2_CS#**/ ISH_SPIA_CS#/GSPI1_CS0#/USB-C_GPP_F17	O	**THC1_SPI2 Chip Select**: Used to select the touch devices if it is connected to THC1_SPI2 interface.
GPP_E13/**THC0_SPI1_IO0**/ GSPI0_MOSI/I2C4_SCL/USB-C_GPP_E13	I/O	**THC0_SPI1_IO0**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E12/**THC0_SPI1_IO1**/ GSPI0_MISO/I2C4_SDA/USB-C_GPP_E12	I/O	**THC0_SPI1_IO1**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E01/USB-C_SMLACLK/ **THC0_SPI1_IO2**/USB-C_GPP_E01	I/O	**THC0_SPI1_IO2**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E02/USB-C_SMLADATA/ **THC0_SPI1_IO3**/USB-C_GPP_E02	I/O	**THC0_SPI1_IO3**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F12/GSXDOUT/**THC1_SPI2_IO0**/ ISH_SPIA_MISO/GSPI1_MOSI/I2C5_SCL/ USB-C_GPP_F12	I/O	**THC1_SPI2_IO0**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F13/GSXSLOAD/**THC1_SPI2_IO1**/ ISH_SPIA_MOSI/GSPI1_MISO/I2C5_SDA/ USB-C_GPP_F13	I/O	**THC1_SPI2_IO1**: A bidirectional signal used to support single, dual and quad mode data transfer.

continued...

Signal Name	Type	Description
GPP_F14/GSXDIN/USB-C_SMLCLK/ **THC1_SPI2_IO2**/GSPI0A_MOSI/USB-C_GPP_F14	I/O	**THC1_SPI2_IO2**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_F15/GSXSRESET#/USB-C_SMLDATA/ **THC1_SPI2_IO3**/GSPI0A_MISO/USB-C_GPP_F15	I/O	**THC1_SPI2_IO3**: A bidirectional signal used to support single, dual and quad mode data transfer.
GPP_E06/**THC0_SPI1_RST#**/USB-C_GPP_E06	O	**THC0_SPI1 Reset:** THC0_SPI1 Reset signal from Touch host controller.
GPP_F16/GSXCLK/**THC1_SPI2_RST#**/ GSPI0A_CLK/USB-C_GPP_F16	O	**THC1 SPI2 Reset:** THC1_SPI2 Reset signal from Touch host controller.
GPP_E17/**THC0_SPI1_INT#**/USB-C_GPP_E17	I	**THC0 SPI1 interrupt:** THC0_SPI1 Interrupt signal.
GPP_F18/**THC1_SPI2_INT#**/ GSPI0A_CS0#/USB-C_GPP_F18	I	**THC1 SPI2 interrupt:** THC1_SPI2 Interrupt signal.

41.3 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
THC0_SPI1_IO[0:3]	Pull-up	20 kohm ± 30%	
THC1_SPI2_IO[0:3]	Pull-up	20 kohm ± 30%	

NOTE

The internal pull-up is disabled when RSMRST# is asserted (during reset).

41.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
THC0_SPI1_CLK	Primary	Undriven	Undriven	Undriven
THC1_SPI2_CLK	Primary	Undriven	Undriven	Undriven
THC0_SPI1_CS#	Primary	Undriven	Undriven	Undriven
THC1_SPI2_CS#	Primary	Undriven	Undriven	Undriven
THC0_SPI1_IO[0:3]	Primary	Undriven	Undriven	Undriven
THC1_SPI2_IO[0:3]	Primary	Undriven	Undriven	Undriven
THC0_SPI1_RST#	Primary	Undriven	Undriven	Undriven
THC1_SPI2_RST#	Primary	Undriven	Undriven	Undriven
THC0_SPI1_INT#	Primary	Undriven	Undriven	Undriven
THC1_SPI2_INT#	Primary	Undriven	Undriven	Undriven
Note: 1. During reset refers to when RSMRST# is asserted.				

42.0 Intel® Serial I/O Universal Asynchronous Receiver/Transmitter (UART) Controllers

The Processor implements three independent UART interfaces, UART0, UART1 and UART2. Each UART interface is a 4-wire interface supporting up to 6.25 Mbit/s.

The interfaces can be used in the low-speed, full-speed, and high-speed modes. The UART communicates with serial data ports that conform to the RS-232 interface protocol.

UART2 only implements the UART Host controller and does not incorporate a DMA controller which is implemented for UART0 and UART1. Therefore, UART2 is restricted to operate in PIO mode only.

The UART interfaces support the following features:

- Up to 6.25 Mbit/s Auto Flow Control mode as specified in the 16750 standard
- Transmitter Holding Register Empty (THRE) interrupt mode
- 64-byte TX and 64-byte RX host controller FIFOs
- DMA support with 64-byte DMA FIFO per channel (up to 32-byte burst)
- Functionality based on the 16550 industry standards
- Programmable character properties, such as number of data bits per character (5-8), optional parity bit (with odd or even select) and number of stop bits (1, 1.5, or 2)
- Line break generation and detection
- DMA signaling with two programmable modes
- Prioritized interrupt identification
- Programmable FIFO enable/disable
- Programmable serial data baud rate
- Modem and status lines are independently controlled
- Programmable BAUD RATE supported (baud rate = (serial clock frequency)/(16xdivisor))

NOTES

1. SIR mode is not supported.
2. External read enable signal for RAM wake up when using external RAMs is not supported.

Table 133. Acronyms

Acronyms	Description
DMA	Direct Memory Access
UART	Universal Asynchronous Receiver/Transmitter
LSx	Low speed IO Controller

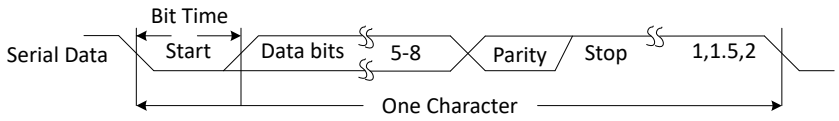
42.1 Functional Description

42.1.1 UART Serial (RS-232) Protocols Overview

Because the serial communication between the UART host controller and the selected device is asynchronous, Start and Stop bits are used on the serial data to synchronize the two devices. The structure of serial data accompanied by Start and Stop bits is referred to as a character.

An additional parity bit may be added to the serial character. This bit appears after the last data bit and before the stop bit(s) in the character structure to provide the UART Host Controller with the ability to perform simple error checking on the received data.

Figure 38. UART Serial Protocol



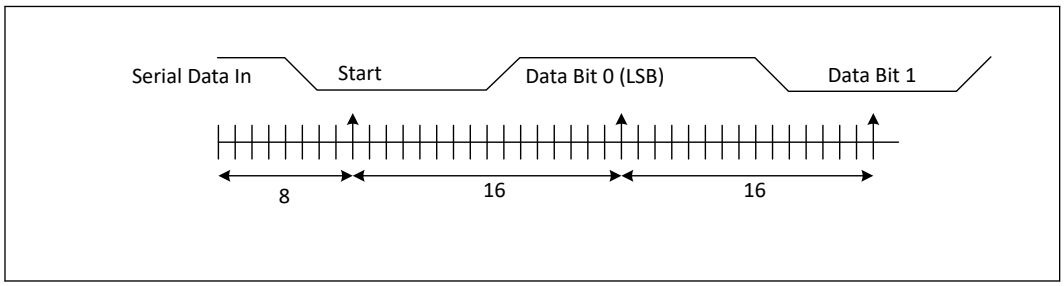
The UART Host Controller Line Control Register (LCR) is used to control the serial character characteristics. The individual bits of the data word are sent after the Start bit, starting with the least significant bit (LSB). These are followed by the optional parity bit, followed by the Stop bit(s), which 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 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 39. UART Receiver Serial Data Sample Points



42.1.2 16550 8-bit Addressing - Debug Driver Compatibility

NOTE

The Processor UART host controller is not compatible with legacy UART 16550 debug-port drivers. The UART host controller operates in 32-bit addressing mode only. UART 16550 legacy drivers only operate with 8-bit addressing. In order to provide compatibility with standard in-box legacy UART drivers a 16550 Legacy Driver mode has been implemented in the UART controller that will convert 8-bit addressed accesses from the 16550 legacy driver to the 32-bit addressing that the UART host controller supports. The UART 16550 8-bit Legacy mode only operates with PIO transactions. DMA transactions are not supported in this mode.

42.1.3 DMA Controller

The UART controllers 0 and 1 (UART0 and UART1) have an integrated DMA controller. Each channel contains a 64-byte FIFO. Max. burst size supported is 32 bytes.

UART controller 2 (UART2) only implements the host controllers and does not incorporate a DMA. Therefore, UART2 is restricted to operate in PIO mode only.

DMA Transfer and Setup Modes

The DMA can operate in the following modes:

1. Memory to peripheral transfers. This mode requires that the peripheral control the flow of the data to itself.
2. Peripheral to memory transfer. This mode requires that the peripheral control the flow of the data from itself.

The DMA supports the following modes for programming:

1. Direct programming. Direct register writes to DMA registers to configure and initiate the transfer.
2. Descriptor based linked list. The descriptors will be stored in memory (such as DDR or SRAM). The DMA will be informed with the location information of the descriptor. DMA initiates reads and programs its own register. The descriptors can form a linked list for multiple blocks to be programmed.
3. Scatter Gather mode

Channel Control

- The source transfer width and destination transfer width are programmable. It can vary to 1 byte, 2 bytes, and 4 bytes.
- Burst size is configurable per channel for source and destination. The number is a power of 2 and can vary between 1,2,4,...,128. this number times the transaction width gives the number of bytes that will be transferred per burst.
- Individual Channel enables. If the channel is not being used, then it should be clock gated.
- Programmable Block size and Packing/Unpacking. Block size of the transfer is programmable in bytes. The block size is not 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.

42.1.4 Reset

Each host controller has an independent rest associated with it. Control of these resets is accessed through the Reset Register.

Each host controller and DMA will be in reset state once powered off and require SW (BIOS or driver) to write into specific reset register to bring the controller from reset state into operational mode.

42.1.5 Power Management

Device Power Down Support

In order to power down peripherals connected to the processorUART bus, the idle, configured state of the I/O signals must be retained to avoid transitions on the bus that can affect the connected powered peripheral. Connected devices are allowed to remain in the D0 active or D2 low power states when the bus is powered off (power gated). The processor HW will prevent any transitions on the serial bus signals during a power gate event.

Latency Tolerance Reporting (LTR)

Latency Tolerance Reporting is used to allow the system to optimize internal power states based on dynamic data, comprehending the current platform activity and service latency requirements. The UART bus architecture, however, does not provide the architectural means to define dynamic latency tolerance messaging. Therefore, the interface supports this by reporting its service latency requirements to the platform power management controller via LTR registers.

The controller’s latency tolerance reporting can be managed by one of the two following schemes. The platform integrator must choose the correct scheme for managing latency tolerance reporting based on the platform, OS and usage.

1. Platform/HW Default Control. This scheme is used for usage models in which the controller's state correctly informs the platform of the current latency requirements. In this scheme, the latency requirement is a function of the controller state. The latency for transmitting data to/from its connected device at a given rate while the controller is active is representative of the active latency requirements. On the other hand if the device is not transmitting or receiving data and idle, there is no expectation for end to end latency.
2. Driver Control. This scheme is used for usage models in which the controller state does not inform the platform correctly of the current latency requirements. If the FIFOs of the connected device are much smaller than the controller FIFOs, or the connected device's end to end traffic assumptions are much smaller than the latency to restore the platform from low power state, driver control should be used.

42.1.6 Interrupts

UART interface has the ability to interrupt and notify the driver that service is required

When an interrupt occurs, the device driver needs to read both the host controller and DMA status and TX completion interrupt registers to identify the interrupt source. Clearing the interrupt is done with the corresponding interrupt register in the host controller or DMA.

42.1.7 Error Handling

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

42.2 Signal Description

Signal Name	Type	Description
GPP_H08/**UART0_RXD**/ USB-C_GPP_H08	I	**UART 0 Receive Data**
GPP_H09/**UART0_TXD**/ USB-C_GPP_H09	O	**UART 0 Transmit Data**
GPP_H10/**UART0_RTS#**/ I3C1A_SDA/ISH_GP10A/ USB-C_GPP_H10	O	**UART 0 Request to Send**
GPP_H11/**UART0_CTS#**/ I3C1A_SCL/ISH_GP11A/ USB-C_GPP_H11	I	**UART 0 Clear to Send**
GPP_H06/I2C3_SDA/ **UART1_RXD**/ ISH_UART1A_RXD/USB-C_GPP_H06	I	**UART 1 Receive Data**
GPP_H07/I2C3_SCL/ **UART1_TXD**/ ISH_UART1A_TXD/USB-C_GPP_H07	O	**UART 1 Transmit Data**

continued...

Signal Name	Type	Description
GPP_H14/ ISH_UART1_RXD/ **UART1A_RXD**/ ISH_I2C1_SDA/USB-C_GPP_H14	O	**UART 1A Receive Data**
GPP_H15/ ISH_UART1_TXD/ **UART1A_TXD**/ ISH_I2C1_SCL/USB-C_GPP_H15	I	**UART 1A Transmit Data**
GPP_F01/CNV_BRI_RSP/ **UART2_RXD**/USB-C_GPP_F01	I	**UART 2 Receive Data**
GPP_F02/CNV_RGI_DT/ **UART2_TXD**/USB-C_GPP_F02	O	**UART 2 Transmit Data**
GPP_F00/CNV_BRI_DT/ **UART2_RTS#**/USB-C_GPP_F00	O	**UART 2 Request to Send**
GPP_F03/CNV_RGI_RSP/ **UART2_CTS#**/USB-C_GPP_F03	I	**UART 2 Clear to Send**

42.3 Integrated Pull-Ups and Pull-Downs

None.

42.4 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
UART[2:0]_RXD	Primary	Undriven	Undriven	Undriven
UART[2:0]_TXD	Primary	Undriven	Undriven	Undriven
UART2_RTS# **UART0_RTS#**	Primary	Undriven	Undriven	Undriven
UART2_CTS# **UART0_CTS#**	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

42.5 LSx

LSx interface supports Four ports. Each port of the LSx controller has two bi-directional signals configured either as Tx (Output) or Rx (Input). Operating voltage of the LSx interface is 1.8 V. LSx controller is responsible for link initialization/ management of HSIO in the Thunderbolt subsystem.

42.5.1 LSx Signal Description

Signal Name	Type	Description	Availability
GPP_C17/ **TBT_LSX0_RXD**/ DDP0_CTRLDATA/ USB-C_GPP_C17	I	**LSx 0 Receive Data**	H/U-Series Processor
GPP_C16/ **TBT_LSX0_TXD**/ DDP0_CTRLCLK/ USB-C_GPP_C16	O	**LSx 0 Transmit Data**	H/U-Series Processor
GPP_C19/ **TBT_LSX1_RXD**/ DDP1_CTRLDATA/ USB-C_GPP_C19	I	**LSx 1 Receive Data**	H/U-Series Processor
GPP_C18/ **TBT_LSX1_TXD**/ DDP1_CTRLCLK/ USB-C_GPP_C18	O	**LSx 1 Transmit Data**	H/U-Series Processor
GPP_C21/ **TBT_LSX2_RXD**/ DDP2_CTRLDATA/ USB-C_GPP_C21	I	**LSx 2 Receive Data**	H/U -Series Processor
GPP_C20/ **TBT_LSX2_TXD**/ DDP2_CTRLCLK/ USB-C_GPP_C20	O	**LSx 2 Transmit Data**	H/U -Series Processor
GPP_C23/ **TBT_LSX3_RXD**/ DDP3_CTRLDATA/ USB-C_GPP_C23	I	**LSx 3 Receive Data**	H/U -Series Processor
GPP_C22/ **TBT_LSX3_TXD**/ DDP3_CTRLCLK/ USB-C_GPP_C22	O	**LSx 3 Transmit Data**	H/U -Series Processor

42.5.2 Integrated Pull-Ups and Pull-Downs

None.

42.5.3 I/O Signal Planes and States

Signal Name	Power Plane	During Reset[1]	Immediately after Reset[1]	S4/S5
TBT_LSX[0:3]_RXD(U/H)	Primary	Undriven	Undriven	Undriven
TBT_LSX[0:3]_TXD (U/H)	Primary	Undriven	Undriven	Undriven
Note: 1. Reset reference for primary well pins is RSMRST#.				

43.0 Private Configuration Space Port ID

The Processor incorporates a wide variety of devices and functions. The registers within these devices are mainly accessed through the primary interface, such as PCI configuration space and IO/MMIO space. Some devices also have registers that are distributed within the Processor Private Configuration Space at individual endpoints (Target Port IDs) which are only accessible through the Processor Sideband Interface. These Processor Private Configuration Space Registers can be addressed via SBREG_BAR or through SBI Index Data pair programming.

Table 134. Private Configuration Space Register Target Port IDs

Processor Device/Function Type	Target Port ID (hex)
FIA Configuration	20
General Purpose I/O (GPIO) Community 0	D1
General Purpose I/O (GPIO) Community 1	D2
General Purpose I/O (GPIO) Community 3	D3
General Purpose I/O (GPIO) Community 4	D4
General Purpose I/O (GPIO) Community 5	D5
DCI	CC
PCIe Controller #1 (SPA)	01
PCIe Controller #2 (SPB)	02
PCIe Controller #3 (SPC)	03
SATA	34
SMBus	6b
eSPI / SPI	6d
xHCI	cb
CNVi	29
PSF6	06
PSF7	07
PSF8	08
PSF13	0D
PSF14	0E
PSF15	0F
ISH Controller	D0
USB 2.0	3A
UART, I^2C, GSPI	33

continued...

Processor Device/Function Type	Target Port ID (hex)
I^3C	5E
Integrated Clock Controller (ICC)	63
GbE	2D
Real Time Clock (Host)	6C
LSx	CD

44.0 Testability and Monitoring

This section contains information regarding the testability signals that provides access to JTAG, run control, system control, and observation resources.

Table 135. Acronyms

Acronyms	Description
BPK	Baltic Peak
BSDL	Boundary Scan Description Language
DCI	Direct Connect Interface
DbC	Debug Class Devices
DFP	Downward Facing Port, USB Type-C term
2W	2-Wire
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
I/OD	Input/Output Open Drain
Intel® TH	Intel® Trace Hub
JTAG	Joint Test Action Group
UFP	Upstream Facing Port, USB Type-C term

Table 136. References

Specification	Document Number/Location
Specification IEEE Standard Test Access Port and Boundary Scan Architecture	http://standards.ieee.org/findstds/standard/1149.1-2013.html

44.1 Signal Description

Table 137. Testability Signals

Signal Name	Type	Description
Processor JTAG Signals		
PROC_JTAG_TCK	I	**Test Clock Input (TCK):** The test clock input provides the clock for the JTAG test logic.
PROC_JTAG_TMS	I	**Test Mode Select (TMS):** The signal is decoded by the Test Access Port (TAP) controller to control test operations.
PROC_JTAG_TDI	I	**Test Data Input (TDI):** Serial test instructions and data is received by the test logic at TDI.
PROC_JTAG_TDO	O	**Test Data Output (TDO):** TDO is the serial output for test instructions and data from the test logic defined in this standard.

continued...

Signal Name	Type	Description
PROC_JTAG_TRST#	I	**Test Reset(TRST)**: Resets the Test Access Port (TAP) logic. This signal should be driven low during power-on Reset.
DBG_PMODE	O	ITP Power Mode Indicator. This signal is used to transmit processor and power/reset information to the Debugger.
PRDY#	O	**Probe Mode Ready:** PRDY# is a processor output used by debug tools to determine processor debug readiness.
PREQ#	I	**Probe Mode Request:** PREQ# is used by debug tools to request debug operation of the processor.
Boundary Scan Sideband Signals		
GPP_D23/**BSSB_LS0_TX**/USB-C_GPP_D23	I/O	**BSSB_LS_TX:** Boundary Scan Sideband Low Speed Transmit for debug purposes
GPP_D22/**BSSB_LS0_RX**/USB-C_GPP_D22	I/O	**BSSB_LS_RX:** Boundary Scan Sideband Low Speed Receive for debug purposes
Breakpoint and Performance Monitor Signals		
BPM[0]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
BPM[1]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
BPM[2]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
BPM[3]	I/O	**Breakpoint and Performance Monitor Signals(BPM):** Outputs from the processor that indicate the status of breakpoints and programmable counters used for monitoring processor performance.
Boot Halt Signal		
BOOTHALT#	I/O	**Boot Halt :** This signal is used for platform boot halt. Supports 1.8 V only.

44.2 Intel® Processor Trace

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

Intel® VTune™ Amplifier for Systems and the Intel® System Debugger are part of Intel® System Studio 2015 (and newer) product, which includes updates for the new debug and trace features, including Intel® PT and Intel® Trace Hub.

Intel® System Studio 2015 is available for download at https://software.intel.com/en-us/system-studio.

An update to the Linux* performance utility, with support for Intel® PT, is available for download at https://github.com/virtuoso/linux-perf/tree/intel_pt. It requires rebuilding the kernel and the perf utility.

44.3 I/O Signal Planes and States

Table 138. Power Planes and States for Testability Signals

Signal Name	Power Plane	Resistors [3]	During Reset[1]	Immediately after Reset[1]	S4/S5
Processor JTAG signals					
PROC_JTAG_TCK	VCCPRIM_IO	Internal Pull-Down	Driven Low	Driven Low	Driven Low
PROC_JTAG_TMS	VCCPRIM_IO	Internal Pull-Up	Driven High	Driven High	Driven High
PROC_JTAG_TDI	VCCPRIM_IO	Internal Pull-Up	Driven High	Driven High	Driven High
PROC_JTAG_TDO	VCCPRIM_IO	External Pull-Up	Undriven	Undriven	Undriven
PROC_JTAG_TRST#	VCCPRIM_IO	Internal Pull-Down	Driven Low	Driven Low	Driven Low
DBG_PMODE	VCCPRIM_IO	Internal Pull-Up	Driven High	Driven High	Driven High
PRDY#	VCCPRIM_IO	Internal Pull-Up	Driven High	Driven High	Driven High
PREQ#	VCCPRIM_IO	Internal Pull-Up	Driven High	Driven High	Driven High
BSSB_LS0_TX	Primary				
BSSB_LS0_RX	Primary				
BPM[3:0]	Primary	External Pull-Up	Undriven	Undriven	Undriven
BOOTHALT#	Primary				

Notes: 1. Reset reference for primary well pins is RSMRST#.
2. It is strongly recommended to reserve pads for PU\PD resistor in parallel to the internal resistor

45.0 Miscellaneous Signals

45.1 Signal Description

Table 139. Signal Descriptions

Signal Name	Type	Description
GPP_B04/**BK0**/ISH_GP4/SBK0/ USB-C_GPP_B04	OD	**Blink BK 0:** This function provides the blink (or PWM) capability. The blink/PWM frequency and duty cycle is programmable through the PWM Control register. Refer to Volume 2 for details.
GPP_B05/**BK1**/ISH_GP0/SBK1/ USB-C_GPP_B05	OD	**Blink BK 1:** This function provides the blink (or PWM) capability. The blink/PWM frequency and duty cycle is programmable through the PWM Control register. Refer to Volume 2 for details.
GPP_B06/**BK2**/ISH_GP1/SBK2/ USB-C_GPP_B06	OD	**Blink BK 2:** This function provides the blink (or PWM) capability. The blink/PWM frequency and duty cycle is programmable through the PWM Control register. Refer to Volume 2 for details.
GPP_B07/**BK3**/ISH_GP2/SBK3/ USB-C_GPP_B07	OD	**Blink BK 3:** This function provides the blink (or PWM) capability. The blink/PWM frequency and duty cycle is programmable through the PWM Control register. Refer to Volume 2 for details.
GPP_B08/**BK4**/ISH_GP3/SBK4/ USB-C_GPP_B08	OD	**Blink BK 4:** This function provides the blink (or PWM) capability. The blink/PWM frequency and duty cycle is programmable through the PWM Control register. Refer to Volume 2 for details.
GPP_E22/DDPA_CTRLCLK/ **DNX_FORCE_RELOAD**/USB-C_GPP_E22	I	**Download and Execute (DnX):** Intel® CSME ROM samples this pin anytime ROM begins execution. This includes the following conditions: • G3 Exit. • Sx, Moff Exit. • Cold Reset (Host Reset with Power Cycle) Exit. • Warm Reset (Host Reset without Power Cycle) Exit if Intel® CSME was shut down in Warm Reset. • 0 => No DnX; 1 => Enter DnX Mode. *Note:* This pin must not be sampled high at the sampling time for normal operation.
GPP_E00/**SATAXPCIE0**/ SATAGP0/USB-C_GPP_E00	I	**SATA port 0 or PCIe port mux select** : This is used to select SATA/PCIe function to support implementations like SATA Express or mSATA.
GPP_F10/**SATAXPCIE1**/ SATAGP1/ISH_GP6A/USB-C_GPP_F10	I	**SATA port 1 or PCIe port mux select** : This is used to select SATA/PCIe function to support implementations like SATA Express or mSATA.
GPP_B04/BK0/ISH_GP4/**SBK0**/ USB-C_GPP_B04	OD	**Serial Blink SBK 0:** This function provides the capability to serialize POST or other messages on the pin to a serial monitor. The Serial Blink message is programmed through the Serial Blink Command/Status and Serial Blink Data registers.
GPP_B05/BK1/ISH_GP0/**SBK1**/ USB-C_GPP_B05	OD	**Serial Blink SBK 1:** This function provides the capability to serialize POST or other messages on the pin to a serial monitor. The Serial Blink message is programmed through the Serial Blink Command/Status and Serial Blink Data registers.

continued...

Signal Name	Type	Description
GPP_B06/BK2/ISH_GP1/**SBK2**/ USB-C_GPP_B06	OD	**Serial Blink SBK 2:** This function provides the capability to serialize POST or other messages on the pin to a serial monitor. The Serial Blink message is programmed through the Serial Blink Command/Status and Serial Blink Data registers.
GPP_B07/BK3/ISH_GP2/**SBK3**/ USB-C_GPP_B07	OD	**Serial Blink SBK 3:** This function provides the capability to serialize POST or other messages on the pin to a serial monitor. The Serial Blink message is programmed through the Serial Blink Command/Status and Serial Blink Data registers.
GPP_B08/BK4/ISH_GP3/**SBK4**/ USB-C_GPP_B08	OD	**Serial Blink SBK 4:** This function provides the capability to serialize POST or other messages on the pin to a serial monitor. The Serial Blink message is programmed through the Serial Blink Command/Status and Serial Blink Data registers.
GPP_B22/**TIME_SYNC0**/ ISH_GP5/USB-C_GPP_B22	I	**Time Synchronization GPIO 0**: Timed GPIO event for time synchronization for interfaces that do not support time synchronization natively.
SKTOCC#	N/A	**Socket Occupied**: Pulled down directly in the processor package to the ground. System board designers may use this signal to determine if the processor is present for safety purposes, it helps to avoid accidentally applying power to the socket while nothing is installed into the socket. If the customers do not want to use or do not need to use the pin (PKG without socket), they can leave it floating.

45.2 Integrated Pull-Ups and Pull-Downs

Table 140. Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value
SATAXPCIE0	Pull-down	20 kohm
SATAXPCIE1	Pull-down	20 kohm

45.3 Ground and Reserved Signals

The following are the general types of reserved (RSVD) signals and connection guidelines:

- RSVD – these signals should not be connected
- RSVD_TP – these signals should be routed to a test point

Arbitrary connection of these signals to VCC, VDD2, VSS, or to any other signal (including each other) may result in component malfunction or incompatibility with future processors. Refer to the table below.

For reliable operation, always connect unused inputs or bi-directional signals to an appropriate signal level. Unused active high inputs should be connected through a resistor to ground (VSS). Unused outputs may be left unconnected however, this may interfere with some Test Access Port (TAP) functions, complicate debug probing and prevent boundary scan testing. A resistor should be used when tying bi-directional signals to power or ground. When tying any signal to power or ground the resistor can also be used for system testability. Resistor values should be within ±20% of the impedance of the baseboard trace.

Table 141. GND, RSVD, and NCTF Signals

Signal Name	Description
VSS	**Ground:** Processor ground node
RSVD	**Reserved**: All signals that are RSVD should not be connected on the board.
RSVD_TP	**Test Point:** Intel recommends to route each RSVD_TP to an accessible test point. Intel may require these test points for platform specific debug. Leaving these test points inaccessible could delay debug by Intel.