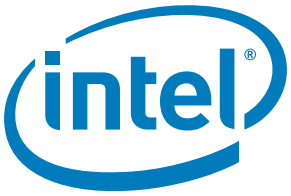


Cherry Trail SoC

External Design Specification (EDS) (Volume 1 of 2)

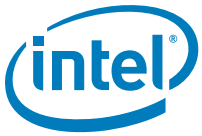
For Volume 2 of 2 refer Document ID: 543698

June 2014

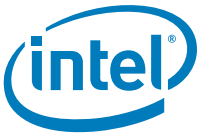
Revision 1.2

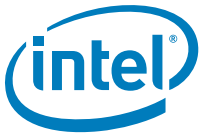
Intel Confidential

Document Number: 539071

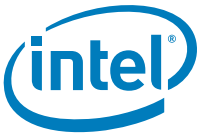


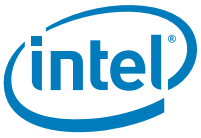
Contents



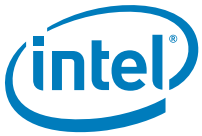


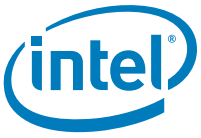
Figures





Tables





Revision History

Document Number	Revision Number	Description	Revision Date
33012	0.5	• Initial release	September 2013
539071	0.6	• This document is de-classified from Intel Restricted Secret to Intel Confidential. Hence, there is a change in the Document Number. • Aligned to SoC Pinlist Rev 0p6 • Changed the GPIO table to accommodate more details and aligned the GPIO table with Pinlist rev 0p6 • Added SoC Pinlist Location and Ballout DDR3L and LPDDR3	October 2013
539071	0.7	• Updated SoC Pinlist Location and Ballout DDR3L and LPDDR3 and Chapter 20, "Electrical Specifications"	November 2013
539071	0.8	• Added SKU table in Chapter 1, "Introduction". - Added Feature List Table 1, "CHT-T SoC Packages" for MSP T4 and VMS T3 - Added Table , "" • Updated Table 3, "Platform Power Well Definitions" • Added Chapter 11, "PCI Express 2.0" - Features, Pinlist and Electrical Specifications • Added Figure 9, "Display Pipe to Port Mapping [MSP-T4]" and Figure 10, "Display Pipe to Port Mapping [VMS-T3]" in Chapter 12, "Graphics, Video and Display". • Updated Data rate in Table 59, "SoC Display Configuration" • Added Chapter 17, "Intel® Sensor Hub". • Chapter 20, "Electrical Specifications" - Updated SDIO AC Specification - Updated SD Card AC Specification - Updated eMMC 4.51 AC Specification - Added I2S AC Specifications - Added SVID AC Specifications - Added SPI AC Specifications - Added SPINOR AC Specifications - Added PMC AC Specifications - Added LPC AC Specifications - Added I2C AC Specifications - Added UART AC Specifications • Added "SoC MSP-T4 Pin List Location" and "SoC VMS-T3 Pin List Location" for MSP T4 and VMS T3 Sku. • Updated Chapter 19.12, "PCU - iLB - GPIO" and added Table 111, "GPIO Signals" • Added Table 187, "SoC Attributes"	December 2013
539071	0.9	• Added Cherry Trail Co-POP details • Updated - Chapter 1, "Introduction" - Chapter 2, "Physical Interfaces" - Chapter 9, "System Memory Controller" - Chapter 14, "USB Controller Interfaces"Section 18.2, "SIO - Serial Peripheral Interface (SPI)"Section 19.4, "PCU - Fast Serial Peripheral Interface (SPI)" - Chapter 20, "Electrical Specifications" • Added Figure 118, Figure 119 and • Added Figure 124, Figure 125, Figure 126,and Figure 127	January 2014

Document Number	Revision Number	Description	Revision Date
539071	1.0	• Updated - Chapter 2, "Physical Interfaces" - Chapter 12, "Graphics, Video and Display" - Chapter 17, "Intel® Sensor Hub" - Section 18.2, "SIO - Serial Peripheral Interface (SPI)"Section 19.4, "PCU - Fast Serial Peripheral Interface (SPI)"Section 19.8, "PCU - iLB - Low Pin Count (LPC) Bridge"Chapter 20, "Electrical Specifications" • Changed LPE_I2S_CLK to 9.6 MHz in Clocking Chapter • Changed eMMC bandwidth to 400 MByte/sec in Storage Chapter • Updated the Reset behavior: Table 41, "Types of Resets" • Updated Table 42 in Chapter 7, "Thermal Management" • Added Figure 6 in Chapter 7, "Thermal Management" • Updated Chapter 22, "Ballout and Ball Map" • Updated Chapter 22.2, "Package Diagrams" • Added Cherry Trail EDS Volume 2: CDI No. 543698 with following registers - SoC Transaction Router - System Memory Controller - Graphics, Video and Display - MIPI-Camera Serial Interface (CSI) and ISP - PCI Express* 2.0 - SoC Storage - USB Controller Interfaces - Serial IO (SIO) - Platform Controller Unit	March 2014

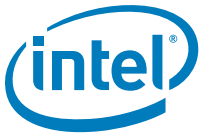
Document Number	Revision Number	Description	Revision Date
539071	1.2	• Chapter 1, "Introduction" - In Table 1, updated "eDP Data Rate" field of MSP-T4 port to 21.6 Gbps. - Updated Table 1 with POR requirements. - Update the USB feature set to match Table 1. - Added note for 4th USB 3.0 port support. - Updated the Package attributes in Table 2. - Added note for Co-POP package under Table 2. • Chapter 2, "Physical Interfaces" - Added SoC GPIO RCOMP table. - Added PWM signal description. - Updated the Package information. - Updated Table 33, "Multiplexed Functions - MSP T4 SoC" - Added Table 34, "Multiplexed Functions - VMS T3 SoC" and Table 35, "Multiplexed Functions - CoPOP SoC" • Chapter 5, "Integrated Clock" - Removed UART clock details from Table 36, "SoC Clock Inputs". - Changed the I2C CLK support - Updated Signal names in the SoC Clock output table • Chapter 7, "Thermal Management" - Edited Figure 6 • Chapter 10, "Graphics, Video and Display" - Changed Embedded DisplayPort Standard Version 1.4b to "v1.3 and v1.4." in Table 63 - Changed to "eDP1.3/eDP1.4" in Table 63. • Chapter 11, "PCI Express 2.0" - Updated the Number of ports. - Updated Figure 16 with lane 1. • Chapter 12, "MIPI-Camera Serial Interface (CSI) and ISP" - Added note in under signal description of MIPI CSI. • Chapter 13, "SoC Storage" - Added MMC1_RCLK data in Signal Description and MMC block diagram Figure 22. - Added HS400 mode support - Updated the data rate for Parallel data line operation - Added SD3_WP signal to the Table 75. • Chapter 14, "USB Controller Interfaces" - Added IVCAM support. - Changed the number of data lanes from [0:4] to [0:3] for USB2 in Signal Description. - Changed the block diagram to appropriate USB ports in Figure 23. - Added xDCI support. • Updated Chapter 15, "Low Power Engine (LPE) for Audio (I^2S)" • Chapter 16, "Intel® Trusted Execution Engine (Intel® TXE)" - Added feature of TXE interaction with NFC • Updated Chapter 17, "Intel® Sensor Hub" to match POR	June 2014

Document Number	Revision Number	Description	Revision Date
539071	1.2	• Chapter 18, "Serial IO (SIO) Overview" - Updated Section 18.2 - Updated to match POR - Updated Section 18.3 - Added section NFC - Added NFC I2C signal description - Added Fast mode plus support for I2C interface • Chapter 19, "Platform Controller Unit (PCU) Overview" - Removed 19.1.1 BIOS/EFI Boot Strap section • Section 19.3, "PCU - Power Management Controller (PMC)" - Removed 25 MHz Clock Support in Platform Clock Support section. • Section 19.5, "PCU - Universal Asynchronous Receiver/ Transmitter (UART)" - Changed Signal names in chapter - Changed IRQ3 to IRQ4 under legacy interrupt • Section 19.7, "PCU - Intel Legacy Block (iLB) Overview" - Changed the S0iX support statement • Section 19.12, "PCU - iLB - GPIO" - Added GPIO controller feature section in GPIO. - Added Section 19.12.4.1 (3.3V versus 1.8V Modes) under GPIO registers. - Removed the GPIO signal table as it was duplicate data already exist in Section 2.25 • Chapter 20, "Electrical Specifications" - Added Tj max/Min and SDP Values in Table 115. - Updated table 117 with correct signal names. - Updated table 118 with VCC/VNN/VGG tolerance values. • In Section 20.7.2, "SVID AC Specification" - Changed Tsu for Data to Min -> 7.5ns - Changed Tco to Min -> -2.0ns • In Section 20.7.4, "LPDDR3 Memory Controller AC Specification" - Changed T_{CTL},T_{DVB}+T_{VDA},T_{CMD} values. • In Section 20.7.5, "Display AC Specifications" - In Table 159 - Changed FAux transaction Frequency to 1 MHz - In Table 157 - Changed the "f_HBR2" field for the "Type" column from 5.1 to 5.4. - Removed Undershoot specification • In Section 20.7.8, "SDIO AC Specification" - Removed Rise/Fall time for 3.3 V operation • In Section 20.7.9, "eMMC 4.51 AC Specification" - Changed CLK Cycle Time (HS200 Mode) to 5 ns - Changed Data/CMD Clock to Output Delay (BC Mode) - Added eMMC_RST# related AC spec • Added Section 20.7.11, "USB 2.0 HSIC AC Specification" • Added Section 20.7.13, "USB SSIC AC Specification" • In Section 20.7.16, "SPI AC Specification" - Changed min/max values Tco of SPI_MOSI - Changed Setup and hold time for CS - Changed setup time for MISO to 10.8ns • In Section 20.7.17, "PCU- Fast SPI AC Specification" - Changed Setup and hold time to 8.8 and 0ns - Changed Tco to 5.2ns max	June 2014

Document Number	Revision Number	Description	Revision Date
		• In Section 20.7.18, "PCU - LPC AC Specification" - Changed Tco --> 12ns max - Changed setup time to 16ns min - Added Trsie/fall time for 3.3V operation - Added 2 more notes for this section • In Section 20.7.19, "I^2C AC Specification" - Changed the Cb (Cap Load) for all modes in Table 185. - Changed the Cb, t_{LOW},t_{HIGH},t_{HD}:DAT, t_r CL, t_f CL, t_f CL1, t_r DA, t_f DA, Values in Table 186. " • In Section 20.7.20, "UART AC Specification" - Changed Rise and fall time -> 0.2 to 1.5ns - Added Sample clk frequency, 15 Mhz • In Section 20.7.14, "I^2S (Audio) AC Specification" - Changed CLK, Tco values for master Mode. - Changed Hold for DATAIN, Setup from FRM, Tco, Rise/Fall time for Slave mode • In Section 20.7.5.5, "MIPI Display Serial Interface (DSI) AC Specification" - Added Note: DSI LP TX slew rates in EDS spec are not measurable with 0 and 5pf load due capacitance of PCB traces.This issue is also mentioned in MIPI Dphy CTS. • Chapter 21, "Ballout and Ball Map" - Updated Figure 116 to Figure 121 - Updated Section 21.2, Section 21.3 and Section 21.4 • Chapter 22, "Package Information" - Updated figures in Section 22.2	

Revision Number Descriptions		
Revision	**Associated Life Cycle Milestone**	**Release Information**
0.5	Design Win Phase	Required Release
0.6–0.7	When Needed	Project Dependent
0.7	Simulations Complete	Required Release
0.8–0.9	When Needed	Project Dependent
1.0	First Silicon Samples	Required Release
1.1–1.4	When Needed	Project Dependent
1.5	Qualification Silicon Samples	Project Dependent
1.6–1.9	When Needed	Project Dependent
NDA - 2.0 Public - XXXXXX-001	First SKU Launch	Required Release Product Launch
2.1 and up	When Needed	Project Dependent

§



1 Introduction

The Cherry Trail SoC is the Intel Architecture (IA) SoC that integrates the next generation Intel® processor core, Graphics, Memory Controller, and I/O interfaces into a single system-on-chip solution.

The figures below shows the system level block diagram of the SoC. Refer to the subsequent chapters for detailed information on the functionality of the different interface blocks.

Note: This document is preliminary. Statements may be missing or subject to change.

The Table 1, “Cherry Trail SoC Packages” lists the different features supported by four Cherry Trail SoC packages.

1.1 Cherry Trail SoC Packages

Table 1. Cherry Trail SoC Packages

Interface	Category	MSP-T4	VMS-T3	Co-POP	MSP-T4 Refresh
CPU	No. of Cores	4	4	4	4
	Burst Speed	2.56 GHz	2 GHz[4]	2.56 GHz	2.56 GHz
Package	Type	17x17mm Type 4	17x17mm Type 3	15x15mm Type4	17x17mm Type 4
	IO count	628	378	378 - Bottom 212 - Top	628
	Ball count	1380	592	1178 - Bottom 396 - Top	1380
	ball pitch	0.4mm	0.65mm	0.4mm	0.4mm
	Z-height	0.937mm	1.002mm	0.721mm[5]	0.937mm
Memory	Interface, Max transfer data rate	Dual Channel 2x64 bit, LPDDR3 -1600MT/S	Single Channel 1x32/64 DDR3L-RS - 1600MT/S	Dual Channel 2x64 bit, LPDDR3 -1600MT/S	Dual Channel 2x64 bit, LPDDR3 -1600MT/S
	Type	BGA	BGA	CO-POP	BGA
PCIe	Number of ports	2	1	2	2
	Port Configuration	1x2, 2x1	x1	1x2, 2x1	1x2, 2x1
Imaging	Number of lanes	6	6	6	6
	Lane configuration	4+2, 3+2, 2+2+2	4+2, 3+2, 2+2	4+2, 2+2+2	4+2, 2+2+2
	Speed	1.5 GHz	1.5 GHz	1.5 GHz	1.5 GHz
	Still & Video	13MP ZSL, 1080p60	5MP, 1080p30	13MP ZSL, 1080p60	13MP ZSL, 1080p60
Media	Media decode rate	H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263	H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263	H.265, H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263	H.265, H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263
	Media encode rate	H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263	H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263	H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263	H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263
Audio	LPE (Low Power Engine)	3 I2S ports	3 I2S ports	3 I2S ports	3 I2S ports

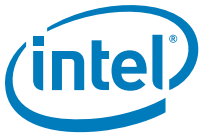


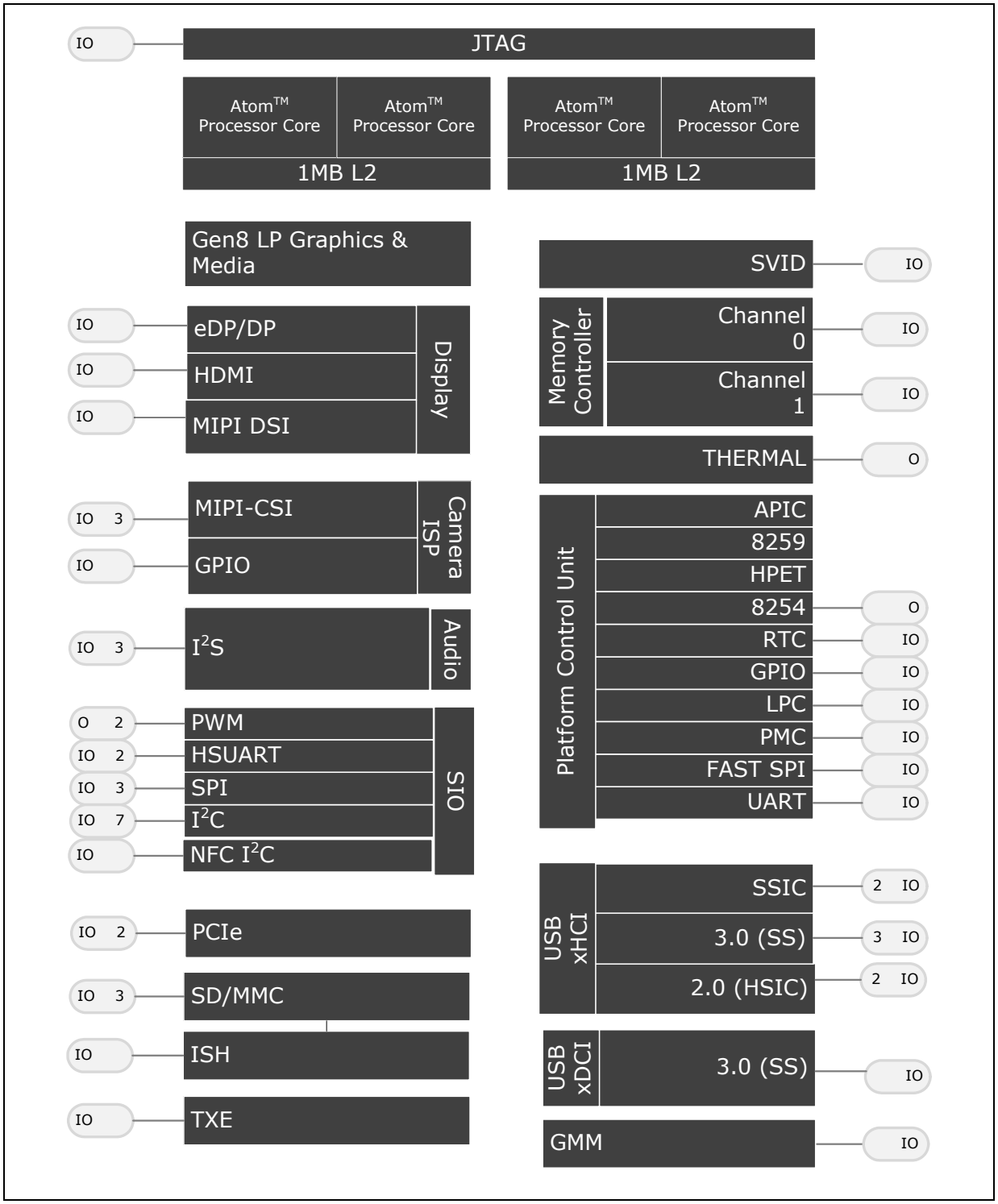
Table 1. Cherry Trail SoC Packages

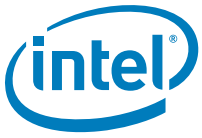
Interface	Category	MSP-T4	VMS-T3	Co-POP	MSP-T4 Refresh
Connectivity & Storage	USB 3.0	3	Not Supported	3	3
	USB 3.0 OTG	1	1	1	1
	USB 2.0	-	3	-	-
	USB SSIC	2	Not Supported	2	2
	USB HSIC	2	2	2	2
	LPC	YES	Not supported	YES	YES
	I2C	7	6	7	7
	I2C Max Speed	1.7 MHz	1.7 MHz	3.4 MHz	3.4 MHz
	I2C NFC	Yes	Yes	Yes	Yes
	SPI	3	Not supported[1]	3	3
	SPI Speed	Master Only up to 25 MHz	Master Only up to 25 MHz	Master Only up to 50MHz	Master Only up to 50 MHz
	Fast SPI	Quad mode	Dual mode	Quad mode	Quad mode
	SD Card	x1 SDR104	x1 SDR104 [2]	x1 SDR104	x1 SDR104
	SDIO	x1 SDR104	x1 SDR104	x1 SDR104	x1 SDR104
	eMMC	4.51	4.51	5.0	5.0
Display	DDI ports	x3	x2	x3	x3
	Max MIPI DSI Resolution	2560x1600	1900x1200	2560x1600	2560x1600
	MIPI-DSI ports	2x 4 Lanes @ 1Ghz	1x 4 Lanes @ 1Ghz	2x 4 Lanes @ 1Ghz	2x 4 Lanes @ 1Ghz
	Max eDP Resolution	2560x1600 @ 24bbp	1920x1200 @ 24bbp	3840x2160 @ 24bbp	3840x2160 @ 24bbp
	eDP ports	2 (2x4 @2.7GHz)	1 (1x4 @2.7GHz)	2 (2x4 @2.7GHz)	2 (2x4 @2.7GHz)
	eDP data Rate	21.6 Gbps	10.8 Gbps	21.6 Gbps	21.6 Gbps
	Max HDMI Resolution	4k x 2k	2k x 1k	4k x 2k	4k x 2k
	MPO	No	No	Yes[3]	Yes[3]

NOTE:

1. One SPI port is multiplexed with reference clock signal which is GPIO signal, and the usage will be dependent on the GPIO configurations on the platform.
2. Is limited to DDR50 due to PMIC power delivery limitation
3. MPO available on Display Pipe B only
4. The Burst Speed mentioned is for 2 Cores.
5. This is PRE-SMT package height. The POST-SMT Z-height is TBD.

Figure 1. SoC Block Diagram





1.2 Terminology

Term	Description
ACPI	Advanced Configuration and Power Interface
Cold Reset	Full reset is when PWROK is de-asserted and all system rails except VCCRTC are powered down
DP	Display Port
DTS	Digital Thermal Sensor
EMI	Electro Magnetic Interference
eDP	embedded Display Port
HDMI	High Definition Multimedia Interface. HDMI supports standard, enhanced, or high-definition video, plus multi-channel digital audio on a single cable. HDMI transmits all Advanced Television Systems Committee (ATSC) HDTV standards and supports 8-channel digital audio, with bandwidth to spare for future requirements and enhancements (additional details available at http://www.hdmi.org/).
Intel® TXE	Intel® Trusted Execution Engine
LPDDR	Low Power Dual Data Rate memory technology.
LPE	Low Power Engine
MIPI CSI	MIPI Camera Interface Specification
MIPI DSI	MIPI Display Interface Specification
MPEG	Moving Picture Experts Group
MSI	Message Signaled Interrupt. MSI is a transaction initiated outside the host, conveying interrupt information to the receiving agent through the same path that normally carries read and write commands.
MSR	Model Specific Register, as the name implies, is model-specific and may change from processor model number (n) to processor model number (n+1). An MSR is accessed by setting ECX to the register number and executing either the RDMSR or WRMSR instruction. The RDMSR instruction will place the 64 bits of the MSR in the EDX: EAX register pair. The WRMSR writes the contents of the EDX: EAX register pair into the MSR.
PWM	Pulse Width Modulation
POSM	Power on state machine
Rank	A unit of DRAM corresponding to the set of SDRAM devices that are accessed in parallel for a given transaction. For a 64-bit wide data bus using 8-bit (x8) wide SDRAM devices, a rank would be eight devices. Multiple ranks can be added to increase capacity without widening the data bus, at the cost of additional electrical loading.
SCI	System Control Interrupt. SCI is used in the ACPI protocol.
SDRAM	Synchronous Dynamic Random Access Memory
SERR	System Error. SERR is an indication that an unrecoverable error has occurred on an I/O bus.
SMC	System Management Controller or External Controller refers to a separate system management controller that handles reset sequences, sleep state transitions, and other system management tasks.

Term	Description
SMI	System Management Interrupt is used to indicate any of several system conditions (such as thermal sensor events, throttling activated, access to System Management RAM, chassis open, or other system state related activity).
SIO	Serial I/O
TMDS	Transition-Minimized Differential Signaling. TMDS is a serial signaling interface used in DVI and HDMI to send visual data to a display. TMDS is based on low-voltage differential signaling with 8/10b encoding for DC balancing.
Warm Reset	Warm reset is when both PMC_PLTRST# and PMC_CORE_PWROK are asserted.

1.3 Feature Overview

1.3.1 Processor Core

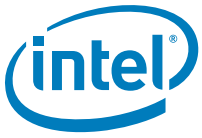
- Up to four IA-compatible low power Intel® processor cores
  - — One thread per core
- Two-wide instruction decode, out of order execution
- On-die, 32 KB 8-way L1 instruction cache and 24 KB 6-way L1 data cache per core
- On-die, 1 MB, 16-way L2 cache, shared per two cores
- 36-bit physical address, 48-bit linear address size support
- Supported C-states: C0, C1, C6C, C6, C7
- Supports Intel® Virtualization Technology (Intel® VT-x2)

1.3.2 System Memory Controller

- Memory Controller supports dual-channel DDR3L-RS/LPDDR3

Note: Cherry Trail CO-POP Package will support PoP (Package on Package) memory devices.

- Up to two ranks per channel (4 ranks in total)
- 32 Bit or 64 Bit data bus.
- ECC supported in single channel mode only for DDR3L-RS.
- Supports DDR3L-RS/LPDDR3 with 1600 MT/s data rate
- Supports x16 and x32 LPDDR3 DRAM device data widths
- Supports x8 and x16 DDR3 DRAM device data widths
- Total memory bandwidth supported is 12.8GB/s (for 1600 MT/s single-channel) to 25.6GB/s (for 1600 MT/s dual-channel)
- Supports different physical mappings of bank addresses to optimize performance
- Supports Dynamic Voltage and Frequency Scaling



- Out-of-order request processing to increase performance
- Aggressive power management to reduce power consumption
- Proactive page closing policies to close unused pages

1.3.3 Display Controller

- Support 3 Display pipes.
- Support 2 MIPI DSI ports
- Support 3 DDI ports to enable eDP 1.4, DP 1.2, DVI, or HDMI 1.4b
- Support 2 panel power sequence for 2 eDP ports
- Supports Audio on DP and HDMI
- Supports Intel® Display Power Saving Technology (DPST) 6.0, Panel Self Refresh (PSR) and Display Refresh Rate Switching Technology (DRRS)

1.3.4 Graphics and Media Engine

- Intel's 8th generation (Gen 8) LP graphics and media encode/decode engine
- Supports 3D rendering, media compositing and video encoding.
- Graphics Burst enabled through energy counters
- Supports DX*11.1, OpenGL 4.3, OGL ES 3.0, OpenCL 1.2.
- 4x anti-aliasing
- Full HW acceleration for decode of HEVC (H.265), H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263 and VP9.
- Full HW acceleration for encode of H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263 and VP9.
- Supports Content protection using PAVP2.0, HDCP 1.4/2.0 and Media Vault DRM.

1.3.5 Image Signal Processor

- Support up to three MIPI CSI ports
- Support for up to 13MP sensors

1.3.6 Power Management

- ACPI 5.0 support
- Processor states: C0, C1, C1E, C6C, C6 and C7
- Display and Graphics device states: D0, D3
- System sleep states: S0, S0ix, S4, S5
- Support CPU and GFx Burst for selected SKUs

- Dynamic I/O power reductions (disabling sense amps on input buffers, tristating output buffers)
- Dynamic memory self-refresh

1.3.7 PCI Express*

- Supports x2 PCIe 2.0 compliant controller
- Supports both Gen1 and Gen2 data rates.
- The controller provides a max data payload of 128B with the capability of splitting the request at 64B granularity.

Supports autonomous up-configuration and autonomous down-configuration as target.

1.3.8 USB Controller

1.3.8.1 USB xHCI Controller

USB Host Controller supports:

- Two (2) Super Speed Inter-Chip (SSIC) port.
- Three (3) Super Speed (SS) ports [Backward Compatible of USB 2.0 HS/FS/LS]
- Two (2) High Speed Inter-Chip (HSIC) ports

Note: SoC can support the 4th SS port when OTG port is in Host mode.

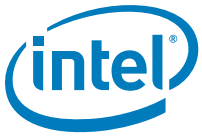
1.3.8.2 USB xDCI Controller

The SoC implements OTG block for device-mode functionality

- Supports one USB 3.0 Super Speed port with backward compatibility of USB 2.0 High Speed and Low/Full Speed.
- Supports SuperSpeed OTG v3.0 device:
- Supports USB3 Debug Device Class Specification [usb3-debug]

1.3.9 Low Power Engine (LPE) Audio Controller

- Support 3 I2S ports.
- I2S and DDI with dedicated DMA
- MP3, AAC, AC3/DD+, WMA9, PCM (WAV)
- Provides HW acceleration for common audio and voice functions such as codecs, acoustic echo cancellation, noise cancellation



1.3.10 Storage

1.3.10.1 Storage Control Cluster (eMMC, SDIO, SD)

- Supports one eMMC 5.0 controller
  - 400 MB/s Data rate
- Supports one SDIO 3.0 interface
  - 800 Mb/s Data rate
- Supports one SDXC controller
  - 800 Mb/s Data rate

1.3.11 Intel® Trusted Execution Engine (Intel® TXE)

Intel TXE is responsible for supporting and handling security related features.

- Supports MediaVault with OMA-DRM and OneTime Password.
- Isolated execution environment for crypto operations
- Supports secure boot - with customer programmable keys to secure code

1.3.12 Serial I/O (SIO)

- Controller for external devices via SPI, UART, I^2C or PWM
- Each port is multiplexed with general purpose I/O for configurations flexibility
- Supports up to 7 I^2C, NFC I^2C, ISH I^2C, 2 HSUART, 2 PWM, 3 SPI interface

1.3.13 Platform Control Unit (PCU)

The platform controller unit is a collection of HW blocks, including UART, debug/boot SPI and Intel legacy block (iLB), that are critical to implement a Windows* compatible platform. Some of its key features are:

- Universal Asynchronous Receiver/Transmitter (UART) with COM1 interface
- A Fast Serial Peripheral Interface (SPI) for Flash only - stores boot FW and system configuration data
- Intel Legacy Block (iLB) supports legacy PC platform features
  - RTC, Interrupts, Timers and Peripheral interface (LPC for TPM) blocks.

1.3.14 Gaussian Mixture Modeling (GMM)*

GMM (Gaussian Mixture Modeling) is an accelerator module designed for speech recognition.

1.3.15 Intel®Sensor Hub

The Intel® Sensor Hub Supports:

- 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 gating and power gating of parts of the ISH together with the ability to turn sensors off.
- The ability to operate independently when the host platform is in low power state

1.3.16 Package

This SoC is packaged in a Flip-Chip Ball Grid Array (FCBGA) package.

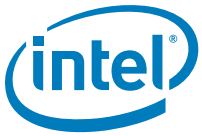
Below table summarizes the package attributes for different SoC Sku's.

Table 2. Package Attributes

	Category	MSP-T4	VMS-T3	Co-POP	MSP-T4 Refresh
Package	Type	17x17mm Type 4	17x17mm Type 3	15x15mm Type4	17x17mm Type 4
	IO count	628	378	378 - Bottom 212 - Top	628
	Ball count	1380	592	1178 - Bottom 396 - Top	1380
	ball pitch	0.4mm	0.65mm	0.4mm	0.4mm
	Z-height	0.937mm	1.002mm	0.721mm[1]	0.937mm

NOTE: This is PRE-SMT package height. The POST-SMT Z-height is TBD.

§



2 Physical Interfaces

Many interfaces contain physical pins. These groups of pins make up the physical interfaces. Because of the large number of interfaces and the small size of the package, Some interfaces share their pins with GPIOs, while others use dedicated physical pins. This chapter summarizes the physical interfaces, including the diversity in GPIO multiplexing options.

2.1 Pin States through Reset

This section describes the states of each signal before, during and directly after reset. Additionally, Some signals have internal pull-up/pull-down termination resistors, and their values are also provided. All signals with the "" symbol are muxed and may not be available without configuration.

Table 3. Platform Power Well Definitions

Power Type	Voltage Range (V)	Power Well Description
VCC0/1	0.75 - 1.15	Variable voltage rail for core
VGG	0.75 - 1.15	Variable voltage rail for Graphics Core
VNN	0.75 - 1.15	Variable voltage rail for SoC.
V1P15	1.15	Fixed voltage rail for SoC, Graphics, camera
V1P05A	1.05	Fixed voltage rail for P-unit, LPE, TXE,I/O's, PLL's and ISH
V1P24	1.24	Fixed voltage rail for I/O's and PLL's.
VDDQ	1.24	Fixed voltage rail for DDR PHY
V1P8A	1.8	Fixed voltage rail for I/O's.
V3P3A	3.3	Fixed voltage rail for I/O's.
V3P3A_V1P8A	1.8/3.3	Fixed voltage rail for SDIO.
V3P3RTC	3.3	Voltage rai For RTC clock.

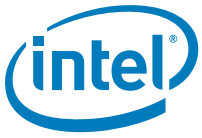
Table 4. Buffer Type Definitions (Sheet 1 of 2)

Buffer Type	Buffer Description
MIPI-DPHY	1.24 V tolerant MIPI DPHY buffer type
USB3 PHY	1.0 V tolerant USB3 PHY buffer type
USB2 PHY	1.8 V tolerant USB3 PHY buffer type
SSIC PHY	1.2 V tolerant SSIC PHY buffer type
HSIC PHY	1.2 V tolerant HSIC PHY buffer type
GPIO	GPIO buffer type. This can be of the following types: 1.8/3.3 V.

Table 4. Buffer Type Definitions (Sheet 2 of 2)

Buffer Type	Buffer Description
MODPHY	1.0 V tolerant MODPHY buffer type
DDR3	1.5 V tolerant DDR3 buffer type
Analog	Analog pins that do not have specific digital requirements. Often used for circuit calibration or monitoring.
GPIOMV, HS	GPIO Buffer type, Medium Voltage(1.8V),High Speed (FMAX~208Mhz)
GPIOMV, MS	GPIO Buffer type, Medium Voltage(1.8V),Medium Speed (FMAX~60Mhz)
GPIOMV, MS, CLK	GPIO Buffer type, Medium Voltage(1.8V),Medium Speed (FMAX~60Mhz), Clock
GPIOMV, HS, CLK	GPIO Buffer type, Medium Voltage(1.8V),High Speed (FMAX~208Mhz), Clock
GPIOMV, HS, RCOMP	GPIO Buffer type, Medium Voltage(1.8V),High Speed (FMAX~208Mhz), RCOMP
GPIOMV, MS, I2C	GPIO Buffer type, Medium Voltage(1.8V),Medium Speed (FMAX~60Mhz), I2C
GPIOHV, HS	GPIO Buffer type, High Voltage(1.8V/3.3V),High Speed (FMAX~208Mhz)
GPIOHV, HS, RCOMP	GPIO Buffer type, High Voltage(1.8V/3.3V),High Speed (FMAX~208Mhz), RCOMP

Note: GPIO mode, where register controlled will not hit FMAX speeds they only matter when functionally used.



2.2 System Memory Controller Interface Signals

Table 5. Default Buffer State Definitions

Buffer State	Description
Z	The SoC places this output in a high-impedance state. For inputs, external drivers are not expected.
Do Not Care	The state of the input (driven or tristated) does not affect the processor. For outputs, it is assumed that the output buffer is in a high-impedance state.
V_{OH}	The SoC drives this signal high with a termination of 50 Ω.
V_{OL}	The SoC drives this signal low with a termination of 50 Ω.
Unknown	The processor drives or expects an indeterminate value.
V_{IH}	The SoC expects/requires the signal to be driven high.
V_{IL}	The SoC expects/requires the signal to be driven low.
"P" 1.1V	USB low speed Single ended 1.
Pull-up	This signal is pulled high by a pull-up resistor (internal or external — internal value specified in “Term” column).
Pull-down	This signal is pulled low by a pull-down resistor (internal or external — internal value specified in “Term” column).
Running	The clock is toggling, or the signal is transitioning.
Off	The power plane for this signal is powered down. The processor does not drive outputs, and inputs should not be driven to the processor. (VSS on output)
1	Buffer drives V_{OH}
0	Buffer drives V_{OL}
H	Buffer Hi Z, weak PU, default to 20K, unless explicitly specified otherwise
L	Buffer Hi Z, weak PD, default to 20K, unless explicitly specified otherwise
Input H	Input enable, weak PU
Output L	Output enable, weak PU
Pgm	Programmable
Retain	retain configuration/data prior to standby

2.2.1 DDR3L-RS

Note: Cherry Trail CO-POP Package will support PoP (Package on Package) memory devices.

Table 6. DDR3L-RS System Memory Signals (Sheet 1 of 2)

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
DDR3_M0_MA[15:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M0_CK[1,0]_P	O	V1P35	DDR	Z	Z	Z
DDR3_M0_CK[1,0]_N	O	V1P35	DDR	Z	Z	Z
DDR3_M0_CKE[3:0]	O	V1P35	DDR	Weak 0	0	0
DDR3_M0_CS[1,0]_N	O	V1P35	DDR	Z	Z	Z
DDR3_M0_CAS_N	O	V1P35	DDR	Z	Z	Z
DDR3_M0_RAS_N	O	V1P35	DDR	Z	Z	Z
DDR3_M0_WE_N	O	V1P35	DDR	Z	Z	Z
DDR3_M0_BS[2:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M0_DRAMRST_N	O	V1P35	DDR	Weak 0	0	1
DDR3_M0_ODT[1,0]	O	V1P35	DDR	Z	Z	Z
DDR3_M0_DQ[63:0]	I/O	V1P35	DDR	Z	Z	Z
DDR3_M0_DM[7:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M0_DQSP[7:0]	I/O	V1P35	DDR	Z	Z	Z
DDR3_M0_DQSN[7:0]	I/O	V1P35	DDR	Z	Z	Z
DDR3_M0_OCAVREF	I	V1P35	DDR	Z	Z	Z
DDR3_M0_ODQVREF	I	V1P35	DDR	Z	Z	Z
DDR3_M0_RCOMPPD	I	V1P35	DDR	Z	Z	Z
DDR3_M1_MA[15:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M1_CK[1,0]_P	O	V1P35	DDR	Z	Z	Z
DDR3_M1_CK[1,0]_N	O	V1P35	DDR	Z	Z	Z
DDR3_M1_CKE[3:0]	O	V1P35	DDR	Weak 0	0	0
DDR3_M1_CS[1,0]_N	O	V1P35	DDR	Z	Z	Z
DDR3_M1_CAS_N	O	V1P35	DDR	Z	Z	Z
DDR3_M1_RAS_N	O	V1P35	DDR	Z	Z	Z
DDR3_M1_WE_N	O	V1P35	DDR	Z	Z	Z
DDR3_M1_BS[2:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M1_DRAMRST_N	O	V1P35	DDR	Weak 0	0	1
DDR3_M1_ODT[1,0]	O	V1P35	DDR	Z	Z	Z
DDR3_M1_DQ[63:0]	I/O	V1P35	DDR	Z	Z	Z
DDR3_M1_DM[7:0]	O	V1P35	DDR	Z	Z	Z
DDR3_M1_DQS[7:0]_P	I/O	V1P35	DDR	Z	Z	Z
DDR3_M1_DQS[7:0]_N	I/O	V1P35	DDR	Z	Z	Z

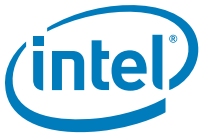


Table 6. DDR3L-RS System Memory Signals (Sheet 2 of 2)

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
DDR3_M1_OCAVREF	I	V1P35	DDR	Z	Z	Z
DDR3_M1_ODQVREF	I	V1P35	DDR	Z	Z	Z
DDR3_M1_RCOMPPD	I	V1P35	DDR	Z	Z	Z
DDR3_DRAM_PWROK	I	V1P35	DDR	Input	Input	Input
DDR3_CORE_PWROK	I	V1P35	DDR	Input	Input	Input

2.2.2 LPDDR3

Table 7. LPDDR3 System Memory Signals (Sheet 1 of 2)

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
LPDDR3_M0_CA[9:0]	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_CK_P_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_CK_N_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_CKE[1:0]_A/B	O	V1P24	DDR	Weak 0	0	0
LPDDR3_M0_CS[1:0]_N	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_ODT_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_DQ[31:0]_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_DM[3:0]_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_DQS[3:0]_P_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_DQS[3:0]_N_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_OCAVREF	I	V1P24	DDR	Z	Z	Z
LPDDR3_M0_ODQVREF	I	V1P24	DDR	Z	Z	Z
LPDDR3_M0_RCOMPPD	I	V1P24	DDR	Z	Z	Z
LPDDR3_M1_CA[9:0]	O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_CK_P_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_CK_N_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_CKE[1:0]_A/B	O	V1P24	DDR	Weak 0	0	0
LPDDR3_M0_CS[1:0]_N	O	V1P24	DDR	Z	Z	Z
LPDDR3_M0_ODT_A/B	O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_DQ[31:0]_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_DM[3:0]_A/B	O	V1P24	DDR	Z	Z	Z

Table 7. LPDDR3 System Memory Signals (Sheet 2 of 2)

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
LPDDR3_M1_DQS[3:0]_P_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_DQS[3:0]_N_A/B	I/O	V1P24	DDR	Z	Z	Z
LPDDR3_M1_OCAVREF	I	V1P24	DDR	Z	Z	Z
LPDDR3_M1_ODQVREF	I	V1P24	DDR	Z	Z	Z
LPDDR3_M1_RCOMPPD	I	V1P24	DDR	Z	Z	Z
LPDDR3_DRAM_PWROK	I	V1P24	DDR	Input	Input	Input
LPDDR3_CORE_PWROK	I	V1P24	DDR	Input	Input	Input

2.3 USB Controller Interface Signals

2.3.1 USB2.0 Interface Signals

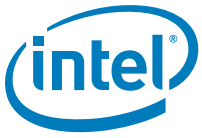
Table 8. USB2.0 Interface Signals

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
USB_DN[3:0]	I/O	V1P8	USB2 PHY	"P" 1.1V	"P" 1.1V	S0i3[1]
USB_DP[3:0]	I/O	V1P8	USB2 PHY	"P" 1.1V	"P" 1.1V	S0i3[1]
USB_OTG_ID	I/O	V1P8	USB2 PHY	Input, weak pull up	Input, weak pull up	Input
USB_VBUSSNS	I/O	V1P8	USB2 PHY	Input	Input	Input
USB_RCOMP	O	V1P8	USB2 PHY	Output	Output	Output
USB_OC[1:0]_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)

NOTES:

— [1]Depends on USB2 Mode

— USB 2.0 Port 0 is the OTG port



2.3.2 USB HSIC Interface Signals

Table 9. USB 2.0 HSIC Interface Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
USB_HSIC0_DATA	I/O	V1P2	HSIC Buffer	Weak 0	Weak 0	Weak 0
USB_HSIC0_STROBE	I/O	V1P2	HSIC Buffer	Weak 1	Weak 1	Weak 1
USB_HSIC1_DATA	I/O	V1P2	HSIC Buffer	Weak 0	Weak 0	Weak 0
USB_HSIC1_STROBE	I/O	V1P2	HSIC Buffer	Weak 1	Weak 1	Weak 1
USB_HSIC_RCOMP	I	V1P2	HSIC Buffer	Z	Z	Z

NOTE: The HSIC should be reset after SoC

2.3.3 USB3.0 Interface Signals

2.3.3.1 USB 3.0 Interface Signals

Table 10. USB 3.0 Interface Signals

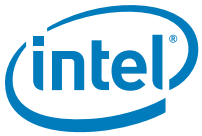
				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
USB3_TXN[3:0]	O	V1P05A	USB3	X	Z	Output
USB3_TXP[3:0]	O	V1P05A	USB3	X	Z	Output
USB3_RXN[3:0]	I	V1P05A	USB3	X	Z	Input
USB3_RXP[3:0]	I	V1P05A	USB3	X	Z	Input
USB3_RCOMP_N	I	V1P05A	USB3	X	Output	Off
USB3_RCOMP_P	I	V1P05A	USB3	X	Output	Off

NOTE: USB3.0 Port 0 is the OTG port

2.3.3.2 USB SSIC Interface Signals

Table 11. USB SSIC Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
USB_SSIC_RX_N[0,1]	I/O	V1P24	SSIC PHY	Input	Input	Input
USB_SSIC_RX_P[0,1]	I/O	V1P24	SSIC PHY	Input	Input	Input
USB_SSIC_TX_N[0,1]	I/O	V1P24	SSIC PHY	Z	Output	Output
USB_SSIC_TX_P[0,1]	I/O	V1P24	SSIC PHY	Z	Output	Output
USB_SSIC_RCOMP_N	O	V1P24	SSIC PHY	Output	Output	Output
USB_SSIC_RCOMP_P	O	V1P24	SSIC PHY	Output	Output	Output



2.4 Integrated Clock Interface Signals

Table 12. Integrated Clock Interface Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
ICLK_OSCIN	I	V1P0	Crystal Oscillator	Input (Crystal)	Input (Crystal)	Input (Crystal)
ICLK_OSCOUT	O	V1P0	Crystal Oscillator	Output (Crystal)	Output (Crystal)	Output (Crystal)
ICLK_ICOMP	O	Analog	Analog	Input	Input	Input
ICLK_RCOMP	O	Analog	Analog	Input	Input	Input

2.5 Display - Digital Display Interface (DDI) Signals

Table 13. Digital Display Interface Signals (Sheet 1 of 2)

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
DDI0_TXP[3:0]	O	V1P24	DDI	Z	Output	Output
DDI0_TXN[3:0]	O	V1P24	DDI	Z	Output	Output
DDI0_AUXP	I/O	V1P24	DDI	Z	Output	Output
DDI0_AUXN	I/O	V1P24	DDI	Z	Output	Output
DDI0_BKLTCTL	I/O	V1P8	GPIOMV, MS	0	0	0
DDI0_BKLTEN	I/O	V1P8	GPIOMV, MS	0	0	0
DDI0_DDC_CLK	I/O	V1P8	GPIOMV, MS, CLK	Input (20k PU)	Input (20k PU)	Input (20k PU)
DDI0_DDC_DATA	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
DDI0_HPD	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	Input (20k PD)
DDI0_VDDEN	I/O	V1P8	GPIOMV, MS	0	0	0
DDI0_RCOMP_N	O	V1P24	DDI	Z	Output	Output
DDI0_RCOMP_P	O	V1P24	DDI	Z	Output	Output
DDI1_TXP[3:0]	O	V1P24	DDI	Z	Output	Output
DDI1_TXN[3:0]	O	V1P24	DDI	Z	Output	Output
DDI1_AUXP	I/O	V1P24	DDI	Z	Output	Output
DDI1_AUXN	I/O	V1P24	DDI	Z	Output	Output

Table 13. Digital Display Interface Signals (Sheet 2 of 2)

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
DDI1_BKLTCTL	I/O	V1P8	GPIOMV, MS	0	0	0
DDI1_BKLTEN	I/O	V1P8	GPIOMV, MS	0	0	0
DDI1_DDC_CLK	I/O	V1P8	GPIOMV, MS, CLK	Input (20k PU)	Input (20k PU)	Input (20k PU)
DDI1_DDC_DATA	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
DDI1_HPD	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	Input (20k PD)
DDI1_VDDEN	I/O	V1P8	GPIOMV, MS	0	0	0
DDI1_RCOMP_N	O	V1P24	DDI	Z	Output	Output
DDI1_RCOMP_P	O	V1P24	DDI	Z	Output	Output
DDI2_DDC_CLK	I/O	V1P8	DDI	Z	Output	Output
DDI2_DDC_DATA	I/O	V1P8	DDI	Z	Output	Output
DDI2_TXP[3:0]	O	V1P24	DDI	Z	Output	Output
DDI2_TXN[3:0]	O	V1P24	DDI	Z	Output	Output
DDI2_AUXP	I/O	V1P24	DDI	Z	Output	Output
DDI2_AUXN	I/O	V1P24	DDI	Z	Output	Output
DDI2_HPD	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	Input (20k PD)

2.6 MIPI DSI Interface Signals

Table 14. MIPI DSI Interface Signals (Sheet 1 of 2)

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
MDSI_A_CLKN	O	V1P24	MIPI-DPHY	0	0	0
MDSI_A_CLKP	O	V1P24	MIPI-DPHY	0	0	0
MDSI_A_DN[3:0]	I/O	V1P24	MIPI-DPHY	0	0	0
MDSI_A_DP[3:0]	I/O	V1P24	MIPI-DPHY	0	0	0
MDSI_C_CLKN	O	V1P24	MIPI-DPHY	0	0	0
MDSI_C_CLKP	O	V1P24	MIPI-DPHY	0	0	0

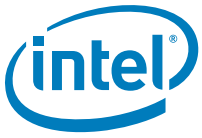


Table 14. MIPI DSI Interface Signals (Sheet 2 of 2)

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
MDSI_C_DN[3:0]	I/O	V1P24	MIPI-DPHY	0	0	0
MDSI_C_DP[3:0]	I/O	V1P24	MIPI-DPHY	0	0	0
MDSI_RCOMP	I/O	V1P24	MIPI-DPHY	0	0	0

2.7 MIPI Camera Serial Interface (CSI) and ISP Interface Signals

Table 15. MIPI CSI Interface Signals

Signal Name	Dir	Plat. Power	Type	Default Buffer State		
				Pwrgood Assert State	Resetout Deassert State	S0ix
MCSI_1_CLKN	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_1_CLKP	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_1_DN[0:3]	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_1_DP[0:3]	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_2_CLKN	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_2_CLKP	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_2_DN[0:1]	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_2_DP[0:1]	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_3_CLKN	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_3_CLKP	I	V1P24	MIPI-DPHY	Input	Input	Input
MCSI_RCOMP	I/O	V1P24	MIPI-DPHY	Input	Input	Input

2.8 PCI Express Signals

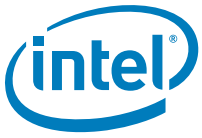
Table 16. PCIe Signals and Clocks

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
PCIE_RXN[0:1]	I	V1P05	PCIe PHY	X	Weak Pull Down	Input
PCIE_RXP[0:1]	I	V1P05	PCIe PHY	X	Weak Pull Down	Input
PCIE_TXN[0:1]	O	V1P05	PCIe PHY	X	Z	Output
PCIE_TXP[0:1]	O	V1P05	PCIe PHY	X	Z	Output
P_RCOMP_N	IO	X	PCIe PHY	X		Off
P_RCOMP_P	IO	X	PCIe PHY	X		Off
PCIE_CLKREQ[0:1]_N	IO	V1P8	GPIOMV, MS	X	Input (20k PU)	Prg

2.9 Low Power Engine (LPE) for Audio (I^2S) Interface Signals

Table 17. LPE Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
LPE_I2S[2:0]_CLK	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	0
LPE_I2S[2:0]_FRM	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	1
LPE_I2S[2:0]_DATAOUT	I/O	V1P8	GPIOMV, MS	0 (20k PD)	0 (20k PD)	0
LPE_I2S[2:0]_DATAIN	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	Input



2.10 Storage Interface Signals

2.10.1 Storage Controller (eMMC, SDIO, SD)

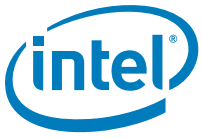
Table 18. Storage Controller (eMMC, SDIO, SD) Interface Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
MMC1_D[7:0]	I/O	V1P8	GPIOMV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
MMC1_CMD	I/O	V1P8	GPIOMV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
MMC1_CLK	I/O	V1P8	GPIOMV, HS, CLK	0 (20k PD)	0 (20k PD)	0 (20k PD)
MMC1_RCLK	I/O	V1P8	GPIOMV, HS	Z (20k PD)	Z	Z
MMC1_RESET_N	I/O	V1P8	GPIOMV, HS	Z	Z	Z
MMC1_RCOMP	I/O	V1P8	GPIOMV, HS, RCOMP	Z	Z	Z
SD2_D[2:0]	I/O	V1P8	GPIOMV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
SD2_D[3]_CD_N	I/O	V1P8	GPIOMV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
SD2_CMD	I/O	V1P8	GPIOMV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
SD2_CLK	I/O	V1P8	GPIOMV, HS, CLK	0 (20k PD)	0	0
SD3_D[3:0]	I/O	V1P8/ V3P3	GPIOHV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
SD3_CMD	I/O	V1P8/ V3P3	GPIOHV, HS	Z (20k PU)	Z (20k PU)	Z (20k PU)
SD3_PWREN_N	I/O	V1P8	GPIOMV, HS	1 (20k PD)	1	-
SD3_CLK	I/O	V1P8/ V3P3	GPIOHV, HS, CLK	0 (20k PD)	0	0
SD3_RCOMP	I/O	V1P8/ V3P3	GPIOHV, HS, RCOMP	Z	Z	Z
SD3_1P8_EN	I/O	V1P8	GPIOMV, HS	0 (20k PD)	0	-
SD3_CD_N	I/O	V1P8	GPIOMV, HS	Input (20k PU)	Input (20k PU)	Input (20k PU)

2.11 High Speed UART Interface Signals

Table 19. High Speed UART Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
UART1_DATAIN	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
UART1_DATAOUT	I/O	V1P8	GPIOMV, MS	1 (20k PU)	1	1
UART1_RTS_N	I/O	V1P8	GPIOMV, MS	1 (20k PU)	1	1
UART1_CTS_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
UART2_DATAIN	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
UART2_DATAOUT	I/O	V1P8	GPIOMV, MS	1 (20k PU)	1	1
UART2_RTS_N	I/O	V1P8	GPIOMV, MS	1 (20k PU)	1	1
UART2_CTS_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)



2.12 I²C Interface Signals

Table 20. I²C Interface Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
I2C0_DATA	I/O	V1P8	GPIOMV, MS, I2C	Z (1k PU, OD)	Z (1k PU, OD)	Z (1k PU, OD)
I2C0_CLK	I/O	V1P8	GPIOMV, MS, I2C	Z (1k PU, OD)	Z (1k PU, OD)	Z (1k PU, OD)
I2C1_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C1_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C2_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C2_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C3_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C3_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C4_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C4_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C5_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C5_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C6_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
I2C6_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)

2.13 NFC I2C Interface Signals

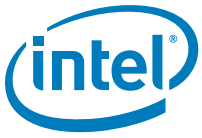
Table 21. NFC I2C Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
NFC_I2C_DATA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
NFC_I2C_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
GPIO_ALERT	I/O	V1P8	GPIOMV, MS	0 (20k PU)	0	0

2.14 PCU- Fast Serial Peripheral Interface (SPI) Signals

Table 22. PCU- Fast Serial Peripheral Interface (SPI) Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
FST_SPI_CLK	I/O	V1P8	GPIOMV, HS	0 (20k PU)	Output	Output
FST_SPI_CS[0]_N	I/O	V1P8	GPIOMV, HS	1 (20k PU)	Output	Output
FST_SPI_CS[1]_N	I/O	V1P8	GPIOMV, HS	Input (20k PU)	Output	Output
FST_SPI_CS[2]_N	I/O	V1P8	GPIOMV, HS	1 (20k PU)	Output	Output
FST_SPI_D[3:0]	I/O	V1P8	GPIOMV, HS	Input (20k PU)	Input (20k PU)	Input (20k PU)



2.15 PCU - Real Time Clock (RTC) Interface Signals

Table 23. PCU - Real Time Clock (RTC) Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
RTC_X1	I	V3P3	RTC PHY	Input (Crystal)	Input (Crystal)	Input (Crystal)
RTC_X2	O	V3P3	RTC PHY	Output (Crystal)	Output (Crystal)	Output (Crystal)
RTC_RST_N	I	V3P3	RTC PHY	Input	Input	Input
RTC_TEST_N	I	V3P3	RTC PHY	Input	Input	Input
RTC_EXTPAD	O	V3P3	RTC PHY	Input	Input	Input

2.16 PCU - Low Pin Count (LPC) Bridge Interface Signals

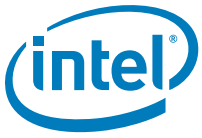
Table 24. PCU - LPC Bridge Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
LPC_AD[3:0]	I/O	V3P3/ V1P8	GPIOHV, HS	Input (20k PU)	Input (20k PU)	Input (20k PU)
LPC_FRAME_N	I/O	V3P3/ V1P8	GPIOHV, HS	1 (20k PU)	1	1
LPC_SERIRQ	I/O	V1P8	GPIOHV, HS	Input (20k PU)	Input (20k PU)	Input (20k PU)
LPC_CLKRUN_N	I/O	V3P3/ V1P8	GPIOHV, HS	Input (20k PU)	Input (20k PU)	Input (20k PU)
LPC_CLKOUT[0]	I/O	V3P3/ V1P8	GPIOHV, HS	0 (20k PU)	Clock	0
LPC_CLKOUT[1]	I/O	V3P3/ V1P8	GPIOHV, HS	Input (20k PD)	Input	Input
LPC_RCOMP	I/O	V3P3/ V1P8	GPIOHV, HS	Z	Z	Z

2.17 PCU - Power Management Controller (PMC) Interface Signals

Table 25. PCU - Power Management Controller (PMC) Interface Signals

				Default Buffer State		
Signal Name	Dir	Plat. Power	Type	Pwrgood Assert State	Resetout Deassert State	S0ix
PMC_PLTRST_N	I/O	V1P8	GPIOMV, MS	0 (20k PU)	1	1
PMC_PWRBTN_N	I	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
PMC_RSTBTN_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
PMC_SUSPWRDNACK	I/O	V1P8	GPIOMV, MS	0 (20k PD)	0 (20k PD)	0 (20k PD)
PMC_SUS_STAT_N	I/O	V1P8	GPIOMV, MS	0 (20k PU)	1	0
PMC_SUSCLK[0]	I/O	V1P8	GPIOMV, MS	0 (20k PD)	32KHz Clock	32KHz Clock
PMC_SLP_S4_N	I/O	V1P8	GPIOMV, MS	0 (20k PU)	1	1
PMC_SLP_S0ix_N	I/O	V1P8	GPIOMV, MS	0 (20k PU)	1	0 at S0ix2
PMC_ACPRESENT	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Input (20k PD)	Input (20k PD)
PMC_BATLOW_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
PMC_WAKE_N	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Input (20k PU)	Input (20k PU)
PMC_CORE_PWROK	I	V3P3	RTC PHY	Input	Input	Input
PMC_RSMRST_N	I	V3P3	RTC PHY	Input	Input	Input



2.18 Serial Peripheral Interface (SPI) Signals

Table 26. Serial Peripheral Interface (SPI) Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
SPI[1,2,3]_CLK	I/O	V1P8	GPIOMV, HS	0 (20k PU)	0	0
SPI[1,2,3]_CS[0:1]_N	I/O	V1P8	GPIOMV, HS	1 (20k PU)	1	1
SPI[1,2,3]_MOSI	I/O	V1P8	GPIOMV, HS	0 (20k PU)	0	0
SPI[1,2,3]_MISO	I/O	V1P8	GPIOMV, HS	Input (20k PU)	Input (20k PD)	Input

2.19 JTAG Interface Signals

Table 27. JTAG Interface Signals

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
JTAG_TCK	I/O	V1P8	GPIOMV, MS	Input (5k PD)	Input (5k PD)	Input (5k PD)
JTAG_TDI	I/O	V1P8	GPIOMV, MS	Input (5k PU)	Input (5k PU)	Input (5k PU)
JTAG_TDO	I/O	V1P8	GPIOMV, MS	Z	Z	Z
JTAG_TMS	I/O	V1P8	GPIOMV, MS	Input (5k PU)	Input (5k PU)	Input (5k PU)
JTAG_TRST_N	I/O	V1P8	GPIOMV, MS	Input (5k PU)	Input (5k PU)	Input (5k PU)
JTAG_PRDY_N	I/O	V1P8	GPIOMV, MS	Z (5k PU, OD)	Output (5k PU, OD)	Z (5k PU, OD)
JTAG_PREQ_N	I/O	V1P8	GPIOMV, MS	Input (5k PU, OD)	Input (5k PU, OD)	Input (5k PU, OD)

2.20 Integrated Sensor Hub Interface Signals

Table 28. Integrated Sensor Hub Interface Signals (Sheet 1 of 2)

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
ISH_GPIO[7:0]	I/O	V1P8	GPIOMV, MS	Input (20k PD)	Z (20k PU)	Z (20k PU)
ISH_GPIO[8]	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)

Table 28. Integrated Sensor Hub Interface Signals (Sheet 2 of 2)

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
ISH_GPIO[9]	I/O	V1P8	GPIOMV, MS	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
ISH_I2C1_SDA	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)
ISH_I2C1_CLK	I/O	V1P8	GPIOMV, MS, I2C	Input (20k PU)	Z (20k PU, OD)	Z (20k PU, OD)

2.21 PWM Interface Signals

Table 29. PWM Interface signal

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
PWM[0]	I/O	V1P8	GPIOMV, MS	0 (20k PD)	0	0
PWM[1]	I/O	V1P8	GPIOMV, MS	0 (20k PU)	0	0

2.22 Miscellaneous Signals

Table 30. Miscellaneous Signals and Clocks

				Default Buffer State		
Signal Name	**Dir**	**Plat. Power**	**Type**	**Pwrgood Assert State**	**Resetout Deassert State**	**S0ix**
SVID_DATA	I/O	V1P8	GPIOMV, MS	0	0	1 or Z
SVID_CLK	O	V1P8	GPIOMV, MS	0	1 or Z	1 or Z
SVID_ALERT_N	I	V1P8	GPIOMV, MS	Input	Input	Input
PLT_CLK[0:5]	I/O	V1P8	GPIOMV, MS	0 (20k PD)	Clock (20k PD)	0[1]
PROCHOT_N	I/O	V1P8	GPIOMV, MS	Z	Z	Z

NOTE: '0' in S0i2 or below

2.23 Hardware Straps

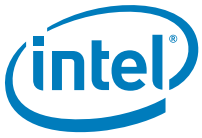
All straps are sampled on the rising edge of **PMC_RSMRST_N**.

Table 31. Straps (Sheet 1 of 2)

Signal Name	Purpose	Pull up/Pull Down	Strap Description
GPIO_SUS[0]	DDI0 Detect	Weak internal pull down of 20K	DDI0 Detect 0 = DDI0 not enabled 1 = DDI0 enabled
GPIO_SUS[1]	DDI1 Detect	Weak internal pull down of 20K	DDI1 Detect 0 = DDI1 not enabled 1 = DDI1 enabled
GPIO_SUS[2]	A16 swap overdrive	Weak internal pull up of 20K	Top Swap (A16 Override) 0 = Change Boot Loader address 1 = Normal Operation
GPIO_SUS[3]	DSI Display Detect	Weak internal pull down of 20K	MIPI DSI Detect 0 = DSI not enabled 1 = DSI enabled
GPIO_SUS[4]	Boot BIOS Strap BBS	Weak internal pull up of 20K	BIOS Boot Selection 0 = Default 1 = SPI

Table 31. Straps (Sheet 2 of 2)

Signal Name	Purpose	Pull up/Pull Down	Strap Description
GPIO_SUS[5]	Flash Descriptor Security Override	Weak internal pull up of 20K	Security Flash Descriptors 0 = Override 1 = Normal Operation
GPIO_SUS[8]	ICLK, USB2, DDI SFR Supply Select	Weak internal pull down of 20K	0 = Supply is 1.25V 1 = Supply is 1.35V This strap also contains PLL LDO 0: supply is 1.25V; 1: supply is 1.35V. Selects supply voltage for LDOs used for PLLs, thermal oscillators, USB2, iCLK and DDI
GPIO_SUS[9]	ICLK, USB2, DDI SFR Bypass	Weak internal pull down of 20K	Bypasses LDOs for ICLK 0 = Use LDOs 1 = Bypass LDOs (Supply1.05V on power pins)
GPIO_SUS[10]	POSM Select	Weak internal pull down of 20K	Selects which POSM (power on state machine) will be observed at time 0 0 = Fuse controller 1 = PMC
GPIO_CAMERASB08	ICLK Xtal OSC Bypass	Weak internal pull down of 20K	0 = No Bypass 1 = Bypass
GPIO_CAMERASB09	CCU SUS RO Bypass	Weak internal pull down of 20K	0 = No Bypass 1 = Bypass
GPIO_CAMERASB11	RTC OSC Bypass	Weak internal pull down of 20K	0 = No Bypass 1 = Bypass



2.24 SoC RCOMP List

Table 32. RCOMP's List

Interface Name	RCOMP Name	Bias	Remarks
DDR3	DDR3_M0_RCOMPPD/ LPDDR3_M0_RCOMPPD	182 ohm ±1% to Ground	RCOMP pins for DDR3
	DDR3_M1_RCOMPPD LPDDR3_M1_RCOMPPD	182 ohm ±1% to Ground	
MIPI DSI	MDSI_RCOMP	150 ohm ±1% to Ground	RCOMP pin for MIPI DSI
MIPI CSI	MCSI_RCOMP	150 ohm ±1% to Ground	RCOMP pin for MIPI CSI
eMMC	MMC1_RCOMP	100 ohm ±1% to Ground	eMMC, SDIO, FST_SPI RCOMP
SD Card	SD3_RCOMP	80.6 ohm ±1% to Ground	SD Card contains its own RCOMP as it can be either 1.8V or 3.3V. Special care is needed to perform an RCOMP any time a card is inserted.
LPC	LPC_RCOMP	100 ohm ±1% to Ground	LPC has its own RCOMP because it can operate at 1.8V or 3.3V
iCLK	ICLK_ICOMP	2.5k ohm ±1% to Ground	The calibration will be handled inside the iCLK
	ICLK_RCOMP	50 ohm ±1% to Ground	
USB2	USB_RCOMP	113 ohm ±1% to Ground	The calibration will be handled inside USB
HSIC	USB_HSIC_RCOMP	50 ohm ±1% to Ground	The calibration is handled inside the USB HSIC.
SSIC	USB_SSIC_RCOMP_P	90 ohm ±1% Between SSIC RCOMP pads	The calibration is handled inside the USB SSIC
	USB_SSIC_RCOMP_N		
USB3	USB3_RCOMP_N	402 ohm 1% between RCOMP pads	The calibration is handled inside the USB3.
	USB3_RCOMP_P		

Table 32. RCOMP's List

GPIO	GPIO0_RCOMP	100 ohm to Ground	Will be shared across all GPIO buffers on the north side of the chip.
PCIE	PCIE_RCOMP_N	402 ohm 1% between RCOMP pads	The Calibration is handled in PCIE.
	PCIE_RCOMP_P		
DDI	DDI0_RCOMP_N	402 ohm 1% between RCOMP pads	The calibration is handled in DDI
	DDI0_RCOMP_P		
	DDI1_RCOMP_N	402 ohm 1% between RCOMP pads	
	DDI1_RCOMP_P		

2.25 GPIO Muxing

Not all interfaces can be active at the same time. To provide flexibility, these shared interfaces are muxed with GPIOs.

Note: All GPIOs default to function as GPIO name at boot. BIOS is responsible for enabling proper configuration.

GPIO Number= GPIO pin location

GPIO mode= GPIO mode in which the pin operates

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 1 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_CAMERASB03	B28	N51		GPIO_CAMERASB03					
JTAG_TMS	B34	N34		JTAG_TMS					
GPIO_DFX1 / C0_BPM1_TX/ C1_BPM1_TX	B38	N3						C0_BPM1_TX	C1_BPM1_TX
PMC_PWRBTN_N	BH10	E8		PMC_PWRBTN_N					
SD3_D[2]	BH18	SE33		SD3_D[2]					

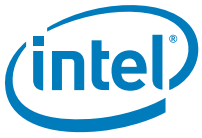


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 2 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PMC_RSTBTN_N	BH24	SE76		PMC_RSTBTN_N					
UART2_RTS_N	BH26	SW19		UART2_RTS_N					
UART2_DATAIN	BH28	SW17		UART2_DATAIN					
LPE_I2S0_CLK	BH32	SW31		LPE_I2S0_CLK					
I2C6_CLK/ NMI_N	BH34	SW53		I2C6_CLK	NMI_N				
I2C2_DATA	BH36	SW62		I2C2_DATA					
PMC_BATLOW_N	BH4	E1		PMC_BATLOW_N					
LPE_I2S2_FRM	BH40	SW96		LPE_I2S2_FRM					
PMC_SUS_STAT_N	BH6	E2		PMC_SUS_STAT_N					
MMC1_CMD	BJ15	SE23		MMC1_CMD					
LPC_FRAME_N/ UART0_DATAIN/ SPI2_MISO	BJ19	SE48		LPC_FRAME_N	UART0_DATAIN		SPI2_MISO		
GPIO_ALERT/ ISH_GPIO[11]/ ISH_UART_DATAIN	BJ21	SE77		GPIO_ALERT	ISH_GPIO[11]	ISH_UART_DATAIN			
FST_SPI_D[2]	BJ25	SW0		FST_SPI_D[2]					
PMC_SLP_S3_N	BJ3	E0		PMC_SLP_S3_N					

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 3 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
LPE_I2S1_DATAIN	BJ30	SW37		LPE_I2S1_DATAIN					
NFC_I2C_CLK	BJ33	SW54		NFC_I2C_CLK					
UART0_DATAIN	BJ37	SW77			UART0_DATAIN				
PMC_PLTRST_N	BJ5	E5		PMC_PLTRST_N					
PMC_WAKE_N	BJ7	E10		PMC_WAKE_N					
PMC_SLP_S4_N	BJ9	E9		PMC_SLP_S4_N					
ISH_GPIO[6]/ I2S4_DATAOUT	BK10	E25		ISH_GPIO[6]		I2S4_DATAOUT			
MMC1_D[3]	BK12	SE26		MMC1_D[3]					
MMC1_D[1]	BK14	SE24		MMC1_D[1]					
SD3_D[0]	BK16	SE35		SD3_D[0]					
SPI1_MOSI	BK18	SE64		SPI1_MOSI					
LPC_CLKOUT[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	BK20	SE51		LPC_CLKOUT[0]	ISH_GPIO[10]		ISH_UART_DATAOUT		
PMC_SUSPWRDNACK	BK22	SE83		PMC_SUSPWRDNACK					
FST_SPI_D[1]	BK26	SW5		FST_SPI_D[1]					
UART2_DATAOUT	BK28	SW21		UART2_DATAOUT					
LPE_I2S0_FRM	BK32	SW35		LPE_I2S0_FRM					

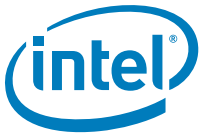


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 4 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
I2C5_CLK	BK34	SW50		I2C5_CLK					
SPI3_MOSI	BK36	SW82			SPI3_MOSI				
LPE_I2S2_CLK	BK38	SW92		LPE_I2S2_CLK					
PMC_ACPRESENT	BK4	E4		PMC_ACPRESENT					
LPE_I2S2_DATAOUT	BK40	SW97		LPE_I2S2_DATAOUT					
ISH_GPIO[9]/ ISH_SPI_MISO/ I2S5_FS	BK8	E20		ISH_GPIO[9]	ISH_SPI_MISO	I2S5_FS			
SD2_CLK	BL11	SE19		SD2_CLK					
SD3_D[3]	BL15	SE32		SD3_D[3]					
SD3_CLK	BL17	SE31		SD3_CLK					
SPI1_MISO	BL19	SE60		SPI1_MISO					
LPC_CLKRUN_N/ UART0_DATAOUT/ SPI2_CLK	BL21	SE46		LPC_CLKRUN_N	UART0_DATAOUT		SPI2_CLK		
FST_SPI_D[3]	BL25	SW3		FST_SPI_D[3]					
UART2_CTS_N	BL27	SW22		UART2_CTS_N					
PMC_SLP_S0IX_N	BL3	E3		PMC_SLP_S0IX_N					
I2C6_DATA/SD3_WP	BL33	SW49		I2C6_DATA	SD3_WP				
I2C2_CLK	BL35	SW66		I2C2_CLK					

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 5 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
UART0_DATAOUT/ SPI3_CLK	BL37	SW79			SPI3_CLK	UART0_DATAOUT			
LPE_I2S2_DATAIN	BL39	SW94		LPE_I2S2_DATAIN					
ISH_GPIO[8]/ I2S5_CLK	BL9	E23		ISH_GPIO[8]		I2S5_CLK			
MMC1_D[0]	BM12	SE17		MMC1_D[0]					
MMC1_D[2]	BM14	SE20		MMC1_D[2]					
ISH_GPIO[7]/ I2S4_DATAIN	BM2	E16		ISH_GPIO[7]		I2S4_DATAIN			
LPC_CLKOUT[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN	BM20	SE49		LPC_CLKOUT[1]	ISH_GPIO[11]		ISH_UART_DATAIN		
LPC_SERIRQ/ SPI2_CS[0]_N	BM24	SE79		LPC_SERIRQ			SPI2_CS[0]_N		
LPE_I2S0_DATAOUT	BM32	SW30		LPE_I2S0_DATAOUT					
SPI3_CS[0]_N	BM38	SW76				SPI3_CS[0]_N			
ISH_GPIO[3]/ I2S3_DATAIN	BM4	E15		ISH_GPIO[3]		I2S3_DATAIN			
PMC_SUSCLK[0]	BM6	E6		PMC_SUSCLK[0]					
ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT	BM8	E26		ISH_I2C1_DATA	ISH_SPI_MOSI	I2S5_DATAOUT			

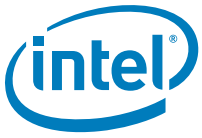


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 6 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
SD2_CMD	BN11	SE22		SD2_CMD					
MMC1_CLK	BN15	SE16		MMC1_CLK					
SPI1_CLK	BN19	SE62		SPI1_CLK					
FST_SPI_D[0]	BN25	SW1		FST_SPI_D[0]					
ISH_I2C1_CLK/ ISH_SPI_CLK/ I2S5_DATAIN	BN3	E17		ISH_I2C1_CLK	ISH_SPI_CLK	I2S5_DATAIN			
NFC_I2C_DATA	BN33	SW51		NFC_I2C_DATA					
SPI3_MISO	BN37	SW81			SPI3_MISO				
ISH_GPIO[1]/I2S3_FS	BN5	E18		ISH_GPIO[1]		I2S3_FS			
SD2_D[3]_CD_N	BP12	SE15		SD2_D[3]_CD_N					
MMC1_D[6]	BP14	SE63		MMC1_D[6]					
SD3_D[1]	BP16	SE30		SD3_D[1]					
LPC_AD[2]/ ISH_GPIO[14]	BP20	SE45		LPC_AD[2]	ISH_GPIO[14]				
USB_OC[0]_N	BP22	SE80		USB_OC[1]_N					
FST_SPI_CLK	BP24	SW2		FST_SPI_CLK					
LPE_I2S1_DATAOUT	BP28	SW34		LPE_I2S1_DATAOUT					
I2C4_CLK/ DDI2_DDC_CLK	BP34	SW52		I2C4_CLK		DDI2_DDC_CLK			
I2C1_CLK	BP36	SW63		I2C1_CLK					
GPIO_SW93	BP38	SW93							

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 7 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
ISH_GPIO[5]/I2S4_FS	BP4	E19		ISH_GPIO[5]		I2S4_FS			
PMC_PLT_CLK[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	BP8	SE0		PMC_PLT_CLK[0]	ISH_GPIO[10]	ISH_UART_DATAOUT			
SD2_D[1]	BR11	SE18		SD2_D[1]					
MMC1_D[4]	BR13	SE67		MMC1_D[4]					
SD3_CMD	BR15	SE34		SD3_CMD					
SPI1_CS[1]_N	BR17	SE66		SPI1_CS[1]_N					
LPC_AD[1]/ ISH_GPIO[13]/ ISH_UART_RTS_N	BR19	SE52		LPC_AD[1]	ISH_GPIO[13]		ISH_UART_RTS_N		
LPC_AD[3]/ ISH_GPIO[15]/ SPI2_MOSI	BR21	SE50		LPC_AD[3]	ISH_GPIO[15]		SPI2_MOSI		
FST_SPI_CS[1]_N	BR23	SW4		FST_SPI_CS[1]_N					
UART1_RTS_N	BR25	SW15		UART1_RTS_N					
UART1_CTS_N	BR27	SW18		UART1_CTS_N					
ISH_GPIO[0]/ I2S3_CLK	BR3	E21		ISH_GPIO[0]		I2S3_CLK			
LPE_I2S1_CLK	BR30	SW32		LPE_I2S1_CLK					
I2C5_DATA	BR33	SW45		I2C5_DATA					
I2C1_DATA	BR35	SW60		I2C1_DATA					

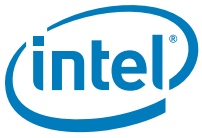


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 8 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
ISH_GPIO[12]/ ISH_UART_CTS_N	BR37	SW75			ISH_GPIO[12]	ISH_UART_CTS_N			
PCIE_CLKREQ[0]_N	BR39	SW90		PCIE_CLKREQ[0]_N					
PMC_PLT_CLK[4]/ ISH_GPIO[14]/ ISH_I2C0_DATA	BR7	SE3		PMC_PLT_CLK[4]	ISH_GPIO[14]	ISH_I2C0_DATA			
PMC_PLT_CLK[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN	BR9	SE2		PMC_PLT_CLK[1]	ISH_GPIO[11]	ISH_UART_DATAIN			
SD2_D[0]	BT10	SE25		SD2_D[0]					
MMC1_D[7]	BT14	SE68		MMC1_D[7]					
SPI1_CS[0]_N	BT18	SE61		SPI1_CS[0]_N					
ISH_GPIO[2]/ I2S3_DATAOUT	BT2	E24		ISH_GPIO[2]		I2S3_DATAOUT			
SD3_1P8_EN	BT22	SE85		SD3_1P8_EN					
UART1_DATAIN/ UART0_DATAIN	BT26	SW16		UART1_DATAIN	UART0_DATAIN				
I2C4_DATA/ DDI2_DDC_DATA	BT32	SW46		I2C4_DATA		DDI2_DDC_DATA			
I2C0_CLK	BT36	SW65		I2C0_CLK					
ISH_GPIO[4]/ I2S4_CLK	BT4	E22		ISH_GPIO[4]		I2S4_CLK			

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 9 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
SD3_WP	BT40	SW95			SD3_WP				
PMC_PLT_CLK[5]/ ISH_GPIO[15]/ ISH_I2C0_CLK	BT6	SE6		PMC_PLT_CLK[5]	ISH_GPIO[15]	ISH_I2C0_CLK			
SD2_D[2]	BU11	SE21		SD2_D[2]					
MMC1_RCLK	BU13	SE69		MMC1_RCLK					
MMC1_D[5]	BU15	SE65		MMC1_D[5]					
SD3_CD_N	BU17	SE81		SD3_CD_N					
LPC_AD[0]/ ISH_GPIO[12]/ ISH_UART_CTS_N	BU19	SE47		LPC_AD[0]	ISH_GPIO[12]		ISH_UART_CTS_N		
USB_OC[1]_N	BU21	SE75		USB_OC[0]_N					
SD3_PWREN_N	BU23	SE78		SD3_PWREN_N					
FST_SPI_CS[0]_N	BU25	SW6		FST_SPI_CS[0]_N					
UART1_DATAOUT/ UART0_DATAOUT	BU27	SW20		UART1_DATAOUT	UART0_DATAOUT				
LPE_I2S0_DATAIN	BU30	SW33		LPE_I2S0_DATAIN					
I2C3_DATA	BU33	SW64		I2C3_DATA					
I2C0_DATA	BU35	SW61		I2C0_DATA					
GPIO_SW78	BU37	SW78							
PCIE_CLKREQ[1]_N	BU39	SW91		PCIE_CLKREQ[1]_N					

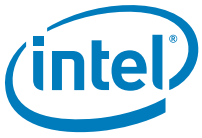


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 10 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PWM[1]/ ISH_GPIO[10]/ ISH_UART_ DATAOUT	BU5	SE1		PWM[1]	ISH_GP IO[10]	ISH_UA RT_DAT AOUT			
PMC_PLT_C LK[2]/ ISH_GPIO[12]/ ISH_UART_ CTS_N	BU7	SE7		PMC_PLT_ CLK[2]	ISH_GP IO[12]	ISH_UA RT_CTS _N			
PMC_PLT_C LK[3]/ ISH_GPIO[13]/ ISH_UART_ RTS_N	BU9	SE4		PMC_PLT_ CLK[3]	ISH_GP IO[13]	ISH_UA RT_RTS _N			
FST_SPI_C S[2]_N	BV24	SW7		FST_SPI_C S[2]_N					
LPE_I2S1_ FRM	BV28	SW36		LPE_I2S1_ FRM					
I2C3_CLK	BV34	SW67		I2C3_CLK					
MMC1_RES ET_N / SPI3_CS[1] _N	BV38	SW80			MMC1_ RESET_ N	SPI3_C S[1]_N			
PWM[0]	BV4	SE5		PWM[0]					
DDI2_HPD	C21	N68		DDI2_HPD					
GPIO_CAM ERASB07	C27	N54		GPIO_CAM ERASB07					
GPIO_CAM ERASB04	C30	N56		GPIO_CAM ERASB04					
SVID_ALER T_N	C33	N38		SVID_ALER T_N					
GPIO_SUS 5/ PMC_SUSC LK[1]	C35	N20		PMC_SUSC LK[1]					

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 11 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_SUS3/ JTAG2_TDI	C37	N17		JTAG2_TDI					
GPIO_DFX2/ ISH_GPIO[13]/ C0_BPM2_TX	C39	N7					ISH_GPIO[13]	C0_BPM2_TX	
GPIO_CAMERASB10	D26	N50		GPIO_CAMERASB10					
SVID_DATA	D32	N33	SVID_DATA						
GPIO_SUS6/ PMC_SUSCLK[2]	D36	N25		PMC_SUSCLK[2]					
GPIO_DFX0/ C0_BPM0_TX	D40	N0						C0_BPM0_TX	
DDI2_DDC_CLK/ MDSI_A_TE/ UART0_DATAOUT	E21	N67		DDI2_DDC_CLK	MDSI_A_TE	UART0_DATAOUT			
DDI1_HPD	E25	N64		DDI1_HPD					
GPIO_CAMERASB06	E27	N49		GPIO_CAMERASB06					
GPIO_CAMERASB02	E30	N46		GPIO_CAMERASB02					
JTAG_TDI	E33	N41		JTAG_TDI					
JTAG_TRST_N	E35	N30		JTAG_TRST_N					
GPIO_SUS0	E37	N15							
GPIO_DFX3/ C0_BPM3_TX	E39	N1						C0_BPM3_TX	

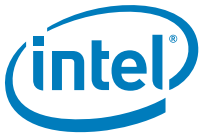


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 12 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
DDI2_DDC_DATA/ MDSI_C_TE/ UART0_DATAIN	F20	N62		DDI2_DDC_DATA	MDSI_C_TE	UART0_DATAIN			
DDI0_VDDEN	F22	N72		DDI0_VDDEN					
DDI0_DDC_CLK/ DDI1_DDC_CLK	F26	N71		DDI0_DDC_CLK	DDI1_DDC_CLK				
GPIO_CAMERASB11	F28	N55		GPIO_CAMERASB11					
SVID_CLK	F32	N40		SVID_CLK					
JTAG_TCK	F34	N31		JTAG_TCK					
GPIO_SUS8	F36	N23		GPIO_SUS8					
GPIO_DFX7/ C0_BPM2_TX	F38	N2						C0_BPM2_TX	
GPIO_CAMERASB05	G27	N45		GPIO_CAMERASB05					
JTAG_PRDY_N	G33	N37		JTAG_PRDY_N					
GPIO_SUS4/ JTAG2_TDO	G37	N22		JTAG2_TDO					
GPIO_DFX5/ C0_BPM0_TX/ C1_BPM0_TX	G39	N4						C0_BPM0_TX	C1_BPM0_TX
DDI1_VDDEN/ MDSI_DDC_DATA	H22	N69		DDI1_VDDEN	MDSI_DDC_DATA				

Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 13 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
DDI0_DDC_DATA/ DDI1_DDC_DATA	H26	N66		DDI0_DDC_DATA	DDI1_DDC_DATA				
PROCHOT_N	H32	N32		PROCHOT_N					
JTAG_PREQ_N	H34	N26		JTAG_PREQ_N					
GPIO_DFX4	H38	N5							
DDI1_BKLTEN/ MDSI_DDC_CLK	J21	N70		DDI1_BKLTEN	MDSI_DDC_CLK				
GPIO_CAMERASB09	J27	N52		GPIO_CAMERASB09					
GPIO_CAMERASB00	J30	N48		GPIO_CAMERASB00					
JTAG_TDO	J33	N39		JTAG_TDO					
GPIO_SUS9	J35	N27		GPIO_SUS9					
GPIO_SUS2/ JTAG2_TMS	J37	N24		JTAG2_TMS					
GPIO_DFX8/ C0_BPM3_TX/ C1_BPM3_TX	J39	N6						C0_BPM3_TX	C1_BPM3_TX
DDI1_BKLTCTL/ MDSI_A_TE/ MDSI_C_TE	K20	N63		DDI1_BKLTCTL		MDSI_A_TE	MDSI_C_TE		
DDI0_BKLTCTL	K22	N65		DDI0_BKLTCTL					
DDI0_HPD	K26	N61		DDI0_HPD					

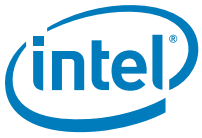


Table 33. Multiplexed Functions - MSP T4 SoC (Sheet 14 of 14)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_CAMERASB08	K28	N47		GPIO_CAMERASB08					
GPIO_CAMERASB01	K32	N53		GPIO_CAMERASB01					
GPIO_SUS10	K34	N16		GPIO_SUS10					
GPIO_SUS7/ PMC_SUSCLK[3]	K36	N18		PMC_SUSCLK[3]					
GPIO_SUS1/ JTAG2_TCK	K38	N19		JTAG2_TCK					
GPIO_DFX6/ C0_BPM1_TX/ C1_BPM1_TX	K40	N8						C0_BPM1_TX	C1_BPM1_TX
DDI0_BKLTEN	L21	N60		DDI0_BKLTEN					

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 1 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
DDI0_DDC_CLK	A10	N71		DDI0_DDC_CLK					
GPIO_SUS9	A13	N27		GPIO_SUS9					
GPIO_SUS0	A14	N15							
DDI0_VDDEN	A9	N72		DDI0_VDDEN					
SD3_CMD	AA10	SE34		SD3_CMD					

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 2 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
FST_SPI_CS[0]_N	AA12	SW6		FST_SPI_CS[0]_N					
LPC_SERIRQ/ SPI2_CS[0]_N	AA13	SE79		LPC_SERIRQ			SPI2_CS[0]_N		
LPE_I2S1_FRM	AA14	SW36		LPE_I2S1_FRM					
LPE_I2S0_DATAOUT	AA15	SW30		LPE_I2S0_DATAOUT					
LPE_I2S0_CLK	AA16	SW31		LPE_I2S0_CLK					
I2C2_CLK	AA17	SW66		I2C2_CLK					
ISH_I2C1_CLK/ ISH_SPI_CLK/ I2S5_DATAIN	AA3	E17		ISH_I2C1_CLK	ISH_SPI_CLK	I2S5_DATAIN			
ISH_GPIO[9]/ ISH_SPI_MISO/ I2S5_FS	AA4	E20		ISH_GPIO[9]	ISH_SPI_MISO	I2S5_FS			
ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT	AA5	E26		ISH_I2C1_DATA	ISH_SPI_MOSI	I2S5_DATAOUT			
MMC1_CMD	AA6	SE23		MMC1_CMD					
MMC1_D[7]	AA7	SE68		MMC1_D[7]					
MMC1_D[0]	AA8	SE17		MMC1_D[0]					
SD3_PWREN_N	AA9	SE78		SD3_PWREN_N					
SD3_D[2]	AB10	SE33		SD3_D[2]					

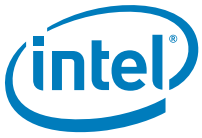


Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 3 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
FST_SPI_D[0]	AB11	SW1		FST_SPI_D[0]					
FST_SPI_CLK	AB12	SW2		FST_SPI_CLK					
UART2_DATAOUT	AB13	SW21		UART2_DATAOUT					
LPE_I2S1_CLK	AB14	SW32		LPE_I2S1_CLK					
LPE_I2S0_DATAIN	AB15	SW33		LPE_I2S0_DATAIN					
I2C5_CLK	AB16	SW50		I2C5_CLK					
I2C4_CLK/ DDI2_DDC_CLK	AB17	SW52		I2C4_CLK		DDI2_DDC_CLK			
I2C4_DATA/ DDI2_DDC_DATA	AB18	SW46		I2C4_DATA		DDI2_DDC_DATA			
LPE_I2S2_CLK	AB19	SW92		LPE_I2S2_CLK					
ISH_GPIO[0]/ I2S3_CLK	AB2	E21		ISH_GPIO[0]		I2S3_CLK			
PCIE_CLKREQ[0]_N	AB20	SW90		PCIE_CLKREQ[0]_N					
ISH_GPIO[2]/ I2S3_DATAOUT	AB3	E24		ISH_GPIO[2]		I2S3_DATAOUT			
PMC_PLT_CLK[3]/ ISH_GPIO[13]/ ISH_UART_RTS_N	AB4	SE4		PMC_PLT_CLK[3]	ISH_GPIO[13]	ISH_UART_RTS_N			
SD2_CMD	AB5	SE22		SD2_CMD					
MMC1_D[2]	AB6	SE20		MMC1_D[2]					

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 4 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
MMC1_D[6]	AB7	SE63		MMC1_D[6]					
SD3_1P8_EN	AB8	SE85		SD3_1P8_EN					
SD3_D[1]	AB9	SE30		SD3_D[1]					
ISH_GPIO[7]/ I2S4_DATAIN	AC1	E16		ISH_GPIO[7]		I2S4_DATAIN			
SD3_D[0]	AC10	SE35		SD3_D[0]					
FST_SPI_D[1]	AC11	SW5		FST_SPI_D[1]					
UART2_RTS_N	AC12	SW19		UART2_RTS_N					
UART1_CTS_N	AC13	SW18		UART1_CTS_N					
UART1_DATAOUT/ UART0_DATAOUT	AC14	SW20		UART1_DATAOUT	UART0_DATAOUT				
NFC_I2C_DATA	AC15	SW51		NFC_I2C_DATA					
NFC_I2C_CLK	AC16	SW54		NFC_I2C_CLK					
I2C2_DATA	AC17	SW62		I2C2_DATA					
GPIO_SW78	AC18	SW78							
LPE_I2S2_FRM	AC19	SW96		LPE_I2S2_FRM					
ISH_GPIO[4]/ I2S4_CLK	AC2	E22		ISH_GPIO[4]		I2S4_CLK			
GPIO_SW93	AC20	SW93							

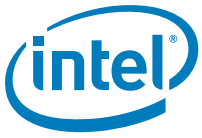


Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 5 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PMC_PLT_CLK[4]/ ISH_GPIO[14]/ ISH_I2C0_DATA	AC3	SE3		PMC_PLT_CLK[4]	ISH_GPIO[14]	ISH_I2C0_DATA			
PMC_PLT_CLK[2]/ ISH_GPIO[12]/ ISH_UART_CTS_N	AC4	SE7		PMC_PLT_CLK[2]	ISH_GPIO[12]	ISH_UART_CTS_N			
MMC1_D[3]	AC6	SE26		MMC1_D[3]					
MMC1_D[4]	AC7	SE67		MMC1_D[4]					
MMC1_D[5]	AC8	SE65		MMC1_D[5]					
SD3_CLK	AC9	SE31		SD3_CLK					
SD3_D[3]	AD10	SE32		SD3_D[3]					
UART2_CTS_N	AD12	SW22		UART2_CTS_N					
UART1_DATAIN/ UART0_DATAIN	AD14	SW16		UART1_DATAIN	UART0_DATAIN				
I2C6_DATA/SD3_WP	AD16	SW49		I2C6_DATA	SD3_WP				
MMC1_RESET_N	AD18	SW80			MMC1_RESET_N				
PWM[1]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	AD2	SE1		PWM[1]	ISH_GPIO[10]	ISH_UART_DATAOUT			
SD3_WP	AD20	SW95			SD3_WP				
PWM[0]	AD3	SE5		PWM[0]					

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 6 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
SD2_D[0]	AD4	SE25		SD2_D[0]					
SD2_CLK	AD5	SE19		SD2_CLK					
SD2_D[3]_CD_N	AD6	SE15		SD2_D[3]_CD_N					
MMC1_CLK	AD8	SE16		MMC1_CLK					
UART2_DATAIN	AE12	SW17		UART2_DATAIN					
UART1_RTS_N	AE13	SW15		UART1_RTS_N					
I2C6_CLK/NMI_N	AE16	SW53		I2C6_CLK	NMI_N				
I2C5_DATA	AE17	SW45		I2C5_DATA					
UART0_DATAIN	AE18	SW77			UART0_DATAIN				
PMC_PLT_CLK[5]/ISH_GPIO[15]/ISH_I2C0_CLK	AE3	SE6		PMC_PLT_CLK[5]	ISH_GPIO[15]	ISH_I2C0_CLK			
SD2_D[1]	AE4	SE18		SD2_D[1]					
SD2_D[2]	AE5	SE21		SD2_D[2]					
MMC1_RCLK	AE8	SE69		MMC1_RCLK					
SD3_CD_N	AE9	SE81		SD3_CD_N					
GPIO_CAMERASB10	B10	N50		GPIO_CAMERASB10					
GPIO_SUS7/PMC_SUSCLK[3]	B12	N18		PMC_SUSCLK[3]					

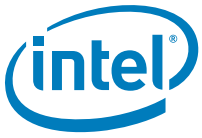


Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 7 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_SUS3/ JTAG2_TDI	B14	N17		JTAG2_TDI					
DDI0_BKLTCTL	B8	N65		DDI0_BKLTCTL					
DDI0_DDC_DATA	C10	N66		DDI0_DDC_DATA					
GPIO_CAMERASB11	C11	N55		GPIO_CAMERASB11					
JTAG_TDI	C12	N41		JTAG_TDI					
GPIO_SUS8	C13	N23		GPIO_SUS8					
GPIO_SUS6/ PMC_SUSCLK[2]	C14	N25		PMC_SUSCLK[2]					
GPIO_SUS1/ JTAG2_TCK	C15	N19		JTAG2_TCK					
GPIO_DFX8/ C0_BPM3_TX/ C1_BPM3_TX	C16	N6						C0_BPM3_TX	C1_BPM3_TX
GPIO_DFX1/ C0_BPM1_TX/ C1_BPM1_TX	C17	N3						C0_BPM1_TX	C1_BPM1_TX
DDI0_BKLTEN	C8	N60		DDI0_BKLTEN					
DDI2_DDC_DATA/ MDSI_C_TE/ UART0_DATAIN	C9	N62		DDI2_DDC_DATA	MDSI_C_TE	UART0_DATAIN			

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 8 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_CAMERASB09	D10	N52		GPIO_CAMERASB09					
GPIO_CAMERASB08	D11	N47		GPIO_CAMERASB08					
GPIO_SUS4/ JTAG2_TDO	D14	N22		JTAG2_TDO					
GPIO_SUS2/ JTAG2_TMS	D15	N24		JTAG2_TMS					
GPIO_DFX6/ C0_BPM1_TX/ C1_BPM1_TX	D16	N8						C0_BPM1_TX	C1_BPM1_TX
GPIO_DFX2/ ISH_GPIO[13]/ C0_BPM2_TX	D17	N7					ISH_GPIO[13]	C0_BPM2_TX	
MDSI_A_TE/ MDSI_C_TE	D8	N63				MDSI_A_TE	MDSI_C_TE		
DDI0_HPD	D9	N61		DDI0_HPD					
DDI2_DDC_CLK/ MDSI_A_TE/ UART0_DATAOUT	E10	N67		DDI2_DDC_CLK	MDSI_A_TE	UART0_DATAOUT			
SVID_ALERT_N	E12	N38		SVID_ALERT_N					
JTAG_TRST_N	E13	N30		JTAG_TRST_N					
JTAG_TCK	E14	N31		JTAG_TCK					

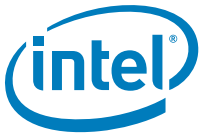


Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 9 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_DFX5/ C0_BPM0_TX/ C1_BPM0_TX	E16	N4						C0_BPM0_TX	C1_BPM0_TX
GPIO_DFX0/ C0_BPM0_TX	E17	N0						C0_BPM0_TX	
DDI2_HPD	E9	N68		DDI2_HPD					
SVID_CLK	F11	N40		SVID_CLK					
SVID_DATA	F12	N33	SVID_DATA						
JTAG_TDO	F13	N39		JTAG_TDO					
JTAG_TMS	F14	N34		JTAG_TMS					
GPIO_DFX7/ C0_BPM2_TX	F16	N2						C0_BPM2_TX	
GPIO_DFX3/ C0_BPM3_TX	F17	N1						C0_BPM3_TX	
GPIO_DFX4	F18	N5							
PROCHOT_N	F9	N32		PROCHOT_N					
LPE_I2S2_DATAIN	U17	SW94		LPE_I2S2_DATAIN					
PMC_SUSPWRDNACK	V12	SE83		PMC_SUSPWRDNACK					
LPC_FRAME_N/ UART0_DATAIN/ SPI2_MISO	V13	SE48		LPC_FRAME_N	UART0_DATAIN		SPI2_MISO		

Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 10 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
LPE_I2S2_DATAOUT	V18	SW97		LPE_I2S2_DATAOUT					
PMC_SUS_STAT_N	V5	E2		PMC_SUS_STAT_N					
I2C1_DATA	W15	SW60		I2C1_DATA					
I2C1_CLK	W16	SW63		I2C1_CLK					
PMC_PWRBTN_N	W3	E8		PMC_PWRBTN_N					
PMC_SLP_S4_N	W4	E3		PMC_SLP_S4_N					
ISH_GPIO[3]/ I2S3_DATAIN	Y1	E15		ISH_GPIO[3]		I2S3_DATAIN			
LPC_CLKRUN_N/ UART0_DATAOUT/ SPI2_CLK	Y12	SE46		LPC_CLKRUN_N	UART0_DATAOUT		SPI2_CLK		
LPE_I2S1_DATAIN	Y13	SW37		LPE_I2S1_DATAIN					
LPE_I2S1_DATAOUT	Y14	SW34		LPE_I2S1_DATAOUT					
LPE_I2S0_FRM	Y16	SW35		LPE_I2S0_FRM					
I2C0_CLK	Y17	SW65		I2C0_CLK					
I2C0_DATA	Y18	SW61		I2C0_DATA					
ISH_GPIO[1]/ I2S3_FS	Y2	E18		ISH_GPIO[1]		I2S3_FS			

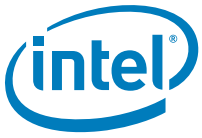


Table 34. Multiplexed Functions - VMS T3 SoC (Sheet 11 of 11)

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PMC_WAKE_N	Y3	E10		PMC_WAKE_N					
PMC_SUSCLK[0]	Y4	E6		PMC_SUSCLK[0]					
PMC_PLTRST_N	Y5	E5		PMC_PLTRST_N					
MMC1_D[1]	Y8	SE24		MMC1_D[1]					
USB_OC[0]_N	Y9	SE80		USB_OC[1]_N					

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
DDI0_BKLTCTL	L17	N65		DDI0_BKLTCTL					
DDI0_BKLTEN	M16	N60		DDI0_BKLTEN					
DDI0_DDC_CLK/ DDI1_DDC_CLK	L9	N71		DDI0_DDC_CLK	DDI1_DDC_CLK				
DDI0_DDC_DATA/ DDI1_DDC_DATA	K6	N66		DDI0_DDC_DATA	DDI1_DDC_DATA				
DDI0_HPD	L5	N61		DDI0_HPD					
DDI0_VDDEN	K16	N72		DDI0_VDDEN					
DDI1_BKLTCTL/ MDSI_A_TE/ MDSI_C_TE	J17	N63		DDI1_BKLTCTL		MDSI_A_TE	MDSI_C_TE		

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
DDI1_BKLTEN/ MDSI_DDC_CLK	F16	N70		DDI1_BKLTEN	MDSI_DDC_CLK				
DDI1_HPD	H6	N64		DDI1_HPD					
DDI1_VDDEN/ MDSI_DDC_DATA	H16	N69		DDI1_VDDEN	MDSI_DDC_DATA				
DDI2_DDC_CLK/ MDSI_A_TE/ UART0_DATAOUT	L11	N67		DDI2_DDC_CLK	MDSI_A_TE	UART0_DATAOUT			
DDI2_DDC_DATA/ MDSI_C_TE/ UART0_DATAIN	N17	N62		DDI2_DDC_DATA	MDSI_C_TE	UART0_DATAIN			
DDI2_HPD	J5	N68		DDI2_HPD					
FST_SPI_CLK	E11	SW2		FST_SPI_CLK					
FST_SPI_CS[0]_N	F18	SW6		FST_SPI_CS[0]_N					
FST_SPI_CS[1]_N	H22	SW4		FST_SPI_CS[1]_N					
FST_SPI_CS[2]_N	C11	SW7		FST_SPI_CS[2]_N					
FST_SPI_D[0]	B12	SW1		FST_SPI_D[0]					
FST_SPI_D[1]	A7	SW5		FST_SPI_D[1]					
FST_SPI_D[2]	E9	SW0		FST_SPI_D[2]					
FST_SPI_D[3]	B6	SW3		FST_SPI_D[3]					

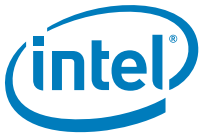


Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_ALERT/ ISH_GPIO[11]/ ISH_UART_DATAIN	D50	SE77		GPIO_ALERT	ISH_GPIO[11]	ISH_UART_DATAIN			
GPIO_CAMERASB00	AJ5	N48		GPIO_CAMERASB00					
GPIO_CAMERASB01	AJ3	N53		GPIO_CAMERASB01					
GPIO_CAMERASB02	AH6	N46		GPIO_CAMERASB02					
GPIO_CAMERASB03	AJ13	N51		GPIO_CAMERASB03					
GPIO_CAMERASB04	AG5	N56		GPIO_CAMERASB04					
GPIO_CAMERASB05	AK14	N45		GPIO_CAMERASB05					
GPIO_CAMERASB06	AJ11	N49		GPIO_CAMERASB06					
GPIO_CAMERASB07	AJ7	N54		GPIO_CAMERASB07					
GPIO_CAMERASB08	AK12	N47		GPIO_CAMERASB08					
GPIO_CAMERASB09	AJ9	N52		GPIO_CAMERASB09					
GPIO_CAMERASB10	AK10	N50		GPIO_CAMERASB10					
GPIO_CAMERASB11	AK8	N55		GPIO_CAMERASB11					

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_DFX 0/ C0_BPM0_ TX	AB12	N0						C0_BPM0 _TX	
GPIO_DFX 1/ C0_BPM1_ TX/ C1_BPM1_ TX	Y6	N3						C0_BPM1 _TX	C1_BPM1_ TX
GPIO_DFX 2/ ISH_GPIO[13]/ C0_BPM2_ TX	AB10	N7					ISH_GPI O[13]	C0_BPM2 _TX	
GPIO_DFX 3/ C0_BPM3_ TX	U5	N1						C0_BPM3 _TX	
GPIO_DFX 4	U3	N5							
GPIO_DFX 5/ C0_BPM0_ TX/ C1_BPM0_ TX	AB8	N4						C0_BPM0 _TX	C1_BPM0_ TX
GPIO_DFX 6/ C0_BPM1_ TX/ C1_BPM1_ TX	AA5	N8						C0_BPM1 _TX	C1_BPM1_ TX
GPIO_DFX 7/ C0_BPM2_ TX	V6	N2						C0_BPM2 _TX	
GPIO_DFX 8/ C0_BPM3_ TX/ C1_BPM3_ TX	AB4	N6						C0_BPM3 _TX	C1_BPM3_ TX
GPIO_SUS 0	AC5	N15							

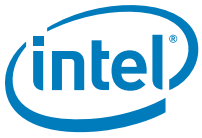


Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
GPIO_SUS1/ JTAG2_TCK	AD10	N19		JTAG2_TCK					
GPIO_SUS10	N5	N16		GPIO_SUS10					
GPIO_SUS2/ JTAG2_TMS	AD12	N24		JTAG2_TMS					
GPIO_SUS3/ JTAG2_TDI	AD2	N17		JTAG2_TDI					
GPIO_SUS4/ JTAG2_TDO	AB6	N22		JTAG2_TDO					
GPIO_SUS5/ PMC_SUSCLK[1]	AD6	N20		PMC_SUSCLK[1]					
GPIO_SUS6/ PMC_SUSCLK[2]	AD8	N25		PMC_SUSCLK[2]					
GPIO_SUS7/ PMC_SUSCLK[3]	AF6	N18		PMC_SUSCLK[3]					
GPIO_SUS8	AE5	N23		GPIO_SUS8					
GPIO_SUS9	AE3	N27		GPIO_SUS9					
SPI3_CS[0]_N	E31	SW76				SPI3_CS[0]_N			
GPIO_SW78	C35	SW78							
SPI3_MISO	F32	SW81			SPI3_MISO				
SPI3_MOSI	E35	SW82			SPI3_MOSI				
GPIO_SW93	E39	SW93							

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
I2C0_CLK	H36	SW65		I2C0_CLK					
I2C0_DATA	D30	SW61		I2C0_DATA					
I2C1_CLK	C27	SW63		I2C1_CLK					
I2C1_DATA	E29	SW60		I2C1_DATA					
I2C2_CLK	E27	SW66		I2C2_CLK					
I2C2_DATA	C23	SW62		I2C2_DATA					
I2C3_CLK	E25	SW67		I2C3_CLK					
I2C3_DATA	E23	SW64		I2C3_DATA					
I2C4_CLK/ DDI2_DDC_CLK	J35	SW52		I2C4_CLK		DDI2_DDC_CLK			
I2C4_DATA/ DDI2_DDC_DATA	K36	SW46		I2C4_DATA		DDI2_DDC_DATA			
I2C5_CLK	F24	SW50		I2C5_CLK					
I2C5_DATA	F30	SW45		I2C5_DATA					
I2C6_CLK/ NMI_N	F28	SW53		I2C6_CLK	NMI_N				
I2C6_DATA/SD3_WP	F26	SW49		I2C6_DATA	SD3_WP				
ISH_GPIO[0]/ I2S3_CLK	AR49	E21		ISH_GPIO[0]		I2S3_CLK			
ISH_GPIO[1]/ I2S3_FS	AV46	E18		ISH_GPIO[1]		I2S3_FS			
ISH_GPIO[12]/ ISH_UART_CTS_N	E33	SW75			ISH_GPIO[12]	ISH_UART_CTS_N			

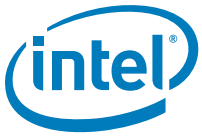


Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
ISH_GPIO[2]/ I2S3_DATAOUT	AP46	E24		ISH_GPIO[2]		I2S3_DATAOUT			
ISH_GPIO[3]/ I2S3_DATAIN	AV48	E15		ISH_GPIO[3]		I2S3_DATAIN			
ISH_GPIO[4]/ I2S4_CLK	AU47	E22		ISH_GPIO[4]		I2S4_CLK			
ISH_GPIO[5]/ I2S4_FS	AT46	E19		ISH_GPIO[5]		I2S4_FS			
ISH_GPIO[6]/ I2S4_DATAOUT	AN47	E25		ISH_GPIO[6]		I2S4_DATAOUT			
ISH_GPIO[7]/ I2S4_DATAIN	AL47	E16		ISH_GPIO[7]		I2S4_DATAIN			
ISH_GPIO[8]/ I2S5_CLK	AL51	E23		ISH_GPIO[8]		I2S5_CLK			
ISH_GPIO[9]/ ISH_SPI_MISO/ I2S5_FS	AL49	E20		ISH_GPIO[9]	ISH_SPI_MISO	I2S5_FS			
ISH_I2C1_CLK/ ISH_SPI_CLK/ I2S5_DATAIN	AR47	E17		ISH_I2C1_CLK	ISH_SPI_CLK	I2S5_DATAIN			
ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT	AM46	E26		ISH_I2C1_DATA	ISH_SPI_MOSI	I2S5_DATAOUT			
JTAG_PRDY_N	R5	N37		JTAG_PRDY_N					

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
JTAG_PREQ_N	T6	N26		JTAG_PREQ_N					
JTAG_TCK	P4	N31		JTAG_TCK					
JTAG_TDI	T8	N41		JTAG_TDI					
JTAG_TDO	U9	N39		JTAG_TDO					
JTAG_TMS	U11	N34		JTAG_TMS					
JTAG_TRST_N	M6	N30		JTAG_TRST_N					
LPC_AD[0] / ISH_GPIO[12]/ ISH_UART_CTS_N	N23	SE47		LPC_AD[0]	ISH_GPIO[12]		ISH_UART_CTS_N		
LPC_AD[1] / ISH_GPIO[13]/ ISH_UART_RTS_N	P28	SE52		LPC_AD[1]	ISH_GPIO[13]		ISH_UART_RTS_N		
LPC_AD[2] / ISH_GPIO[14]	L23	SE45		LPC_AD[2]	ISH_GPIO[14]				
LPC_AD[3] / ISH_GPIO[15]/ SPI2_MOSI	M22	SE50		LPC_AD[3]	ISH_GPIO[15]		SPI2_MOSI		
LPC_CLKOUT[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	J23	SE51		LPC_CLKOUT[0]	ISH_GPIO[10]		ISH_UART_DATAOUT		
LPC_CLKOUT[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN	M28	SE49		LPC_CLKOUT[1]	ISH_GPIO[11]		ISH_UART_DATAIN		

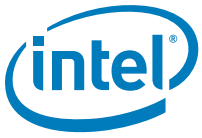


Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
LPC_CLKRUN_N/ UART0_DATAOUT/ SPI2_CLK	K28	SE46		LPC_CLKRUN_N	UART0_DATAOUT		SPI2_CLK		
LPC_FRAME_N/ UART0_DATAIN/ SPI2_MISO	K22	SE48		LPC_FRAME_N	UART0_DATAIN		SPI2_MISO		
LPC_SERIRQ/ SPI2_CS[0]_N	D46	SE79		LPC_SERIRQ			SPI2_CS[0]_N		
LPE_I2S0_CLK	H42	SW31		LPE_I2S0_CLK					
LPE_I2S0_DATAIN	C43	SW33		LPE_I2S0_DATAIN					
LPE_I2S0_DATAOUT	E43	SW30		LPE_I2S0_DATAOUT					
LPE_I2S0_FRM	L35	SW35		LPE_I2S0_FRM					
LPE_I2S1_CLK	E45	SW32		LPE_I2S1_CLK					
LPE_I2S1_DATAIN	F44	SW37		LPE_I2S1_DATAIN					
LPE_I2S1_DATAOUT	B44	SW34		LPE_I2S1_DATAOUT					
LPE_I2S1_FRM	G43	SW36		LPE_I2S1_FRM					
LPE_I2S2_CLK	E37	SW92		LPE_I2S2_CLK					
LPE_I2S2_DATAIN	F38	SW94		LPE_I2S2_DATAIN					
LPE_I2S2_DATAOUT	A31	SW97		LPE_I2S2_DATAOUT					
LPE_I2S2_FRM	L41	SW96		LPE_I2S2_FRM					

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
MMC1_CLK	P46	SE16		MMC1_CLK					
MMC1_CMD	AA47	SE23		MMC1_CMD					
MMC1_D[0]	M46	SE17		MMC1_D[0]					
MMC1_D[1]	W49	SE24		MMC1_D[1]					
MMC1_D[2]	N47	SE20		MMC1_D[2]					
MMC1_D[3]	W47	SE26		MMC1_D[3]					
MMC1_D[4]	L45	SE67		MMC1_D[4]					
MMC1_D[5]	G47	SE65		MMC1_D[5]					
MMC1_D[6]	G49	SE63		MMC1_D[6]					
MMC1_D[7]	F50	SE68		MMC1_D[7]					
MMC1_RCLK	E49	SE69		MMC1_RCLK					
MMC1_RESET_N/ SPI3_CS[1]_N	F36	SW80			MMC1_RESET_N	SPI3_CS[1]_N			
NFC_I2C_CLK	H28	SW54		NFC_I2C_CLK					
NFC_I2C_DATA	H30	SW51		NFC_I2C_DATA					
PCIE_CLKREQ[0]_N	C39	SW90		PCIE_CLKREQ[0]_N					
PCIE_CLKREQ[1]_N	J41	SW91		PCIE_CLKREQ[1]_N					
PMC_ACPRESENT	AC47	E4		PMC_ACPRESENT					
PMC_BATLOW_N	AB48	E1		PMC_BATLOW_N					

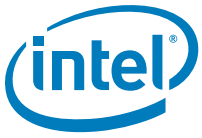


Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PMC_PLT_CLK[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	BA45	SE0		PMC_PLT_CLK[0]	ISH_GPIO[10]	ISH_UART_DATAOUT			
PMC_PLT_CLK[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN	BA43	SE2		PMC_PLT_CLK[1]	ISH_GPIO[11]	ISH_UART_DATAIN			
PMC_PLT_CLK[2]/ ISH_GPIO[12]/ ISH_UART_CTS_N	AT42	SE7		PMC_PLT_CLK[2]	ISH_GPIO[12]	ISH_UART_CTS_N			
PMC_PLT_CLK[3]/ ISH_GPIO[13]/ ISH_UART_RTS_N	AR41	SE4		PMC_PLT_CLK[3]	ISH_GPIO[13]	ISH_UART_RTS_N			
PMC_PLT_CLK[4]/ ISH_GPIO[14]/ ISH_I2C0_DATA	AT40	SE3		PMC_PLT_CLK[4]	ISH_GPIO[14]	ISH_I2C0_DATA			
PMC_PLT_CLK[5]/ ISH_GPIO[15]/ ISH_I2C0_CLK	AT44	SE6		PMC_PLT_CLK[5]	ISH_GPIO[15]	ISH_I2C0_CLK			
PMC_PLTRST_N	AG47	E5		PMC_PLTRST_N					
PMC_PWRBTN_N	AH46	E8		PMC_PWRBTN_N					
PMC_RSTBTN_N	F46	SE76		PMC_RSTBTN_N					

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
PMC_SLP_S0IX_N	AC51	E3		PMC_SLP_S0IX_N					
PMC_SLP_S3_N	AG49	E0		PMC_SLP_S3_N					
PMC_SLP_S4_N	AK46	E9		PMC_SLP_S4_N					
PMC_SUS_STAT_N	AJ47	E2		PMC_SUS_STAT_N					
PMC_SUSCLK[0]	AC49	E6		PMC_SUSCLK[0]					
PMC_SUSPWRDNACK	B48	SE83		PMC_SUSPWRDNACK					
PMC_WAKE_N	AK48	E10		PMC_WAKE_N					
PROCHOT_N	U13	N32		PROCHOT_N					
PWM[0]	AD46	SE5		PWM[0]					
PWM[1]/ ISH_GPIO[10]/ ISH_UART_DATAOUT	AC45	SE1		PWM[1]	ISH_GPIO[10]	ISH_UART_DATAOUT			
SD2_CLK	M50	SE19		SD2_CLK					
SD2_CMD	R49	SE22		SD2_CMD					
SD2_D[0]	L47	SE25		SD2_D[0]					
SD2_D[1]	L49	SE18		SD2_D[1]					
SD2_D[2]	R47	SE21		SD2_D[2]					
SD2_D[3]_CD_N	J47	SE15		SD2_D[3]_CD_N					
SD3_1P8_EN	D48	SE85		SD3_1P8_EN					
SD3_CD_N	C49	SE81		SD3_CD_N					
SD3_CLK	Y46	SE31		SD3_CLK					

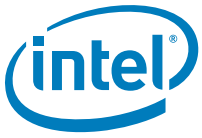


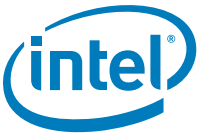
Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
SD3_CMD	T46	SE34		SD3_CMD					
SD3_D[0]	H46	SE35		SD3_D[0]					
SD3_D[1]	U47	SE30		SD3_D[1]					
SD3_D[2]	V46	SE33		SD3_D[2]					
SD3_D[3]	AB46	SE32		SD3_D[3]					
SD3_PWREN_N	C47	SE78		SD3_PWREN_N					
SD3_WP	D38	SW95			SD3_WP				
SPI1_CLK	AK40	SE62		SPI1_CLK					
SPI1_CS[0]_N	AH40	SE61		SPI1_CS[0]_N					
SPI1_CS[1]_N	AH42	SE66		SPI1_CS[1]_N					
SPI1_MISO	AK42	SE60		SPI1_MISO					
SPI1_MOSI	AK44	SE64		SPI1_MOSI					
SVID_ALERT_N	T10	N38		SVID_ALERT_N					
SVID_CLK	N3	N40		SVID_CLK					
SVID_DATA	T12	N33	SVID_DATA						
UART0_DATAIN	F34	SW77			UART0_DATAIN				
UART0_DATAOUT/ SPI3_CLK	C31	SW79			SPI3_CLK	UART0_DATAOUT			
UART1_CTS_N	C19	SW18		UART1_CTS_N					
UART1_DATAIN/ UART0_DATAIN	D22	SW16		UART1_DATAIN	UART0_DATAIN				
UART1_DATAOUT/ UART0_DATAOUT	A23	SW20		UART1_DATAOUT	UART0_DATAOUT				

Table 35. Multiplexed Functions - CoPOP SoC

GPIO Pin Name	Package Ball #	GPIO #	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6
UART1_RTS_N	E21	SW15		UART1_RTS_N					
UART2_CTS_N	F20	SW22		UART2_CTS_N					
UART2_DATAIN	E19	SW17		UART2_DATAIN					
UART2_DATAOUT	B20	SW21		UART2_DATAOUT					
UART2_RTS_N	F22	SW19		UART2_RTS_N					
USB_OC[0]_N	M30	SE80		USB_OC[1]_N					
USB_OC[1]_N	K30	SE75		USB_OC[0]_N					

§



3 Processor Core

Up to four out-of-order execution processor cores are supported, each dual core module supports up to 1 MB of L2 cache.

3.1 Features

- 14nm Process technology
- Quad Out-of-Order Execution (OOE) processor cores
- Primary 32 KB, 8-way L1 instruction cache and 24 KiB, 6-way L1 write-back data cache
- Cores are grouped into dual-core modules: modules share a 1 MB, 16-way L2 cache (2 MB total for Quad Core)Intel® Streaming SIMD Extensions 4.1 and 4.2 (SSE4.1 and SSE4.2), which include new instructions for media and for fast XML parsing
- Intel® 64 Bit architecture
- Support for IA 32-bit
- Support for Intel® VT-x2
- Supports Intel® Advanced Encryption Standard (AES) New instructions (AES-NI)
- Support for Intel® Carry-Less Multiplication Instruction (PCLMULQDQ)
- Support for a Digital Random Number Generator (DRNG)
- Supports C0, C1, C1E, C6C, C6 and C7 states.
- Thermal management support via Intel® Thermal Monitor (TM1 & TM2)
- Uses Power Aware Interrupt Routing (PAIR)

Note: Intel® Hyper-Threading Technology is not supported.

3.1.1 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 and chipsets. Intel® Virtualization Technology for IA-32, Intel® 64 and Intel® Architecture (Intel® VT-x2) added hardware support in the processor to improve the virtualization performance and robustness.

Intel® VT-x2 specifications and functional descriptions are included in the *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B* and is available at: http://www.intel.com/products/processor/manuals/index.htm.

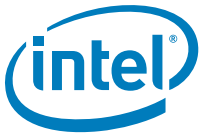
Other Intel® VT-x2 documents can be referenced at: http://www.intel.com/technology/virtualization/index.htm

3.1.1.1 Intel® VT-x2 Objectives

- Robust: VMMs no longer need to use paravirtualization or binary translation. This means that they will be able to run off-the-shelf OSs 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-x2 provides hardware acceleration for virtualization of IA platforms. Virtual Machine Monitor (VMM) can use Intel® VT-x2 features to provide improved reliable virtualized platform.

3.1.1.1.1 Intel® VT-x2 Features

- Extended Page Tables (EPT)
  - EPT is hardware assisted page table physical memory virtualization
  - Support guest VM execution in unpaged protected mode or in real-address mode
  - It eliminates VM exits from guest OS to the VMM for shadow page-table maintenance
- Virtual Processor IDs (VPID)
  - A VM Virtual Processor ID is used to tag processor core hardware structures (such as TLBs) to allow a logic processor to cache information (such as TLBs) for multiple linear address spaces
  - This avoids flushes on VM transitions to give a lower-cost VM transition time and an overall reduction in virtualization overhead
- Guest Preemption Timer
  - Mechanism for a VMM to preempt the execution of a guest OS VM 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 flexibility in guest VM scheduling and building Quality of Service (QoS) schemes
- Descriptor-Table Exiting
  - Descriptor-table exiting allows a VMM to protect a guest OS from internal (malicious software based) attack by preventing 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

- VM Functions
  - A VM function is an operation provided by the processor that can be invoked using the VMFUNC instruction from guest VM without a VM exit
  - A VM function to perform EPTP switching is supported and allows guest VM to load a new value for the EPT pointer, thereby establishing a different EPT paging structure hierarchy

3.1.2 Security and Cryptography Technologies

3.1.2.1 Advanced Encryption Standard New Instructions (AES-NI)

The processor supports Advanced Encryption Standard New Instructions (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). AES-NI are valuable for a wide range of cryptographic applications, for example: applications that perform bulk encryption/decryption, authentication, random number generation, and authenticated encryption. AES is broadly accepted as the standard for both government and industry applications, and is widely deployed in various protocols.

AES-NI consists of six Intel® SSE instructions. Four instructions, namely 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 a full hardware for support AES, offering security, high performance, and a great deal of flexibility.

3.1.2.2 PCLMULQDQ Instruction

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

3.1.2.3 Digital Random Number Generator

The processor introduces a software visible digital random number generation mechanism supported by a high quality entropy source. This capability is available to programmers through the new RDRAND instruction. The resultant random number generation capability is designed to comply with existing industry standards (ANSI X9.82 and NIST SP 800-90).

Some possible uses of the new RDRAND instruction include cryptographic key generation as used in a variety of applications including communication, digital signatures, secure storage, etc.

3.1.3 Power Aware Interrupt Routing

PAIR is an improvement in H/W routing of "redirectable" interrupts. Each core power-state is considered in the routing selection to reduce the power or performance impact of interrupts. System BIOS configures the routing algorithm, e.g. fixed-priority, rotating, hash, or PAIR, during setup via non-architectural register. The PAIR algorithm can be biased to optimize for power or performance and the largest gains will be seen in systems with high interrupt rates.

3.2 Platform Identification and CPUID

In addition to verifying the processor signature, the intended processor platform type must be determined to properly target the microcode update. The intended processor platform type is determined by reading bits [52:50] of the IA32_PLATFORM_ID register, (MSR 17h) within the processor. This is a 64-bit register that must be read using the RDMSR instruction. The 3 Platform Id bits, when read as a binary coded decimal (BCD) number, indicate the bit position in the microcode update header's Processor Flags field that is asSoCiated with the installed processor.

Executing the CPUID instruction with EAX=1 will provide the following information.

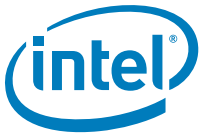
EAX	Field description
[31:28]	Reserved
[27:20]	Extended Family value
[19:16]	Extended Model value
[15:13]	Reserved
[12]	Processor Type Bit
[11:8]	Family value
[7:4]	Model value
[3:0]	Stepping ID Value

3.3 References

For further details of Intel® 64 and IA-32 architectures refer to Intel® 64 and IA-32 Architectures Software Developer's Manual Combined Volumes:1, 2A, 2B, 2C, 3A, 3B, and 3C:

- http://www.intel.com/content/www/µs/en/processors/architectures-software-developer-manuals.html

For more details on AES-NI refer to:



- Intel ® Performance Primitives (IPP) web page - http://software.intel.com/en-us/intel-ipp/
- White Paper on AES-NI - http://software.intel.com/en-us/articles/intel-advanced-encryption-standard-aes-instructions-set/

For more details on using the RDRAND instruction refer to Intel® Advanced Vector Extensions Programming Reference.

§

4 SoC Transaction Router

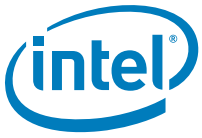
The SoC Transaction Router is a central hub that routes transactions between the CPU cores, graphics controller, IO and the memory controller. In general, it handles:

- CPU Core Interface: Requests for CPU Core-initiated memory and IO read and write operations and processor-initiated message-signaled interrupt transactions
- Device MMIO and PCI configuration routing
- Buffering and memory arbitration
- PCI Config and MMIO accesses to host device (0/0/0)

4.1 Register Map

For more information on SoC Transaction Router registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§



5 Integrated Clock

Clocks are integrated, consisting of multiple variable frequency clock domains, across different voltage domains. This architecture achieves a low power clocking solution that supports the various clocking requirements of the SoC's many interfaces.Platform clocking is provided internally by the iClock block and does not require external devices for clocking. All the required platform clocks are provided by only two inputs: a 19.2 MHz primary reference for the integrated clock block and a 32.768 kHz reference for the Real Time Clock (RTC) block. Both of these would likely be implemented as crystal references.

The different inputs and outputs are listed below.

Table 36. SoC Clock Inputs

Clock Domain	Signal Name	Frequency	Usage / Description
Main	ICLK_OSCIN ICLK_OSCOUT	19.2 MHz	Reference crystal for the iCLK PLL
RTC	RTC_X1 RTC_X2	32.768 kHz	RTC crystal I/O for RTC block
LPC	LPC_CLKOUT	19.2 MHz	Can be configured as an input to compensate for board routing delays through Soft Strap.

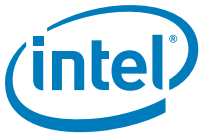
Table 37. SoC Clock Outputs (Sheet 1 of 2)

Clock Domain	Signal Name	Frequency	Usage / Description
DDR	DDR3_M0_CKP[1,0] DDR3_M0_CKN[1,0] DDR3_M1_CKP[1,0] DDR3_M1_CKN[1,0]	800 MHz	Drives the Memory ranks 0-1. Data rate (MT/s) is 2x the clock rate.
SDXC	MMC1_CLK SD2_CLK SD3_CLK	200 MHz	Clock for Storage Devices
SPI	SPI1_CLK FST_SPI_CLK	20 MHz, 33 MHz, 50 MHz	Clock for SPI flash
PMIC/COMMS	PMC_SUSCLK[0]	32.768 kHz	Pass through clock from RTC oscillator
LPC	LPC_CLKOUT[0:1]	19.2 MHz	Provided to devices requiring LPC clock
Display Port	DDI[0]_TXP[3] DDI[0]_TXN[3]	162 or 270 MHz	Differential clock for DP devices
HDMI	DDI[2]_TXP[3] DDI[2]_TXN[3]	25-297 MHz	Differential clock for HDMI devices
HDMI DDC	DDI[2:0]_DDCCLK	100 kHz	Clock for HDMI DDC devices

Table 37. SoC Clock Outputs (Sheet 2 of 2)

Clock Domain	Signal Name	Frequency	Usage / Description
MIPI DSI	MDSI_A_CLKP MDSI_A_CLKN MDSI_C_CLKP MDSI_C_CLKN	1000 MHz	Differential clock for MIPI DSI Devices
MIPI CSI	MCSI1_CLKP MCSI1_CLKN MCSI2_CLKP MCSI2_CLKN MCSI3_CLKP MCSI3_CLKN	200-400 MHz	Clocks for front and rear cameras
SVID	SVID_CLK	20 MHz	Clock used by voltage regulator
I^2S	LPE_I2S[2:0]_CLK	9.6 MHz	Continuous serial clock for I^2S interfaces
Platform Clocks	PLT_CLK [5:0]	19.2MHz	Platform clocks.
SIO SPI	SPI_CLK	15 MHz	SPI clock output
I^2C	I2C[6:0]_CLK	1.7MHz	I^2C clocks
NFC	NFC_I2C_CLK	100 kHz	Clock for NFC device

§



6 Power Up and Reset Sequence

This chapter provides information on the following topics:

- "Power Up Sequences"
- "Power Down Sequences"
- "Reset Behavior"

6.1 SoC System States

6.1.1 System Sleeping States Control (S-states)

The SoC supports the S0, S0i1, S0i2, S0i3, S4, and S5 sleep states. S4 and S5 states are identical from a hardware and power perspective. The differentiation is software determined (S4 = Suspend to Disk).

The SoC platform architecture assumes the usage of an external power management controller e.g., CPLD or PMIC. Some flows in this section refer to the power management controller for support of the S-states transitions.

The SoC sleep states are described in Chapter 9, "Power Management".

6.2 Power Up Sequences

6.2.1 RTC Power Well Transition (G5 to G3 States Transition)

When RTC_VCC (Real Time Clock power) is applied via RTC battery, the following occurs (see Figure 2 for timing):

1. RTC_VCC ramps. RTC_RST_N should be low.
2. The system starts the real time clock oscillator.
3. A minimum of t1 units after RTC_VCC ramps, the external RTC RC circuit de-asserts RTC_RST_N. The system is now in the G3 state. RTC oscillator is unlikely to be stable at this point.

Figure 2. RTC Power Well Timing Diagrams

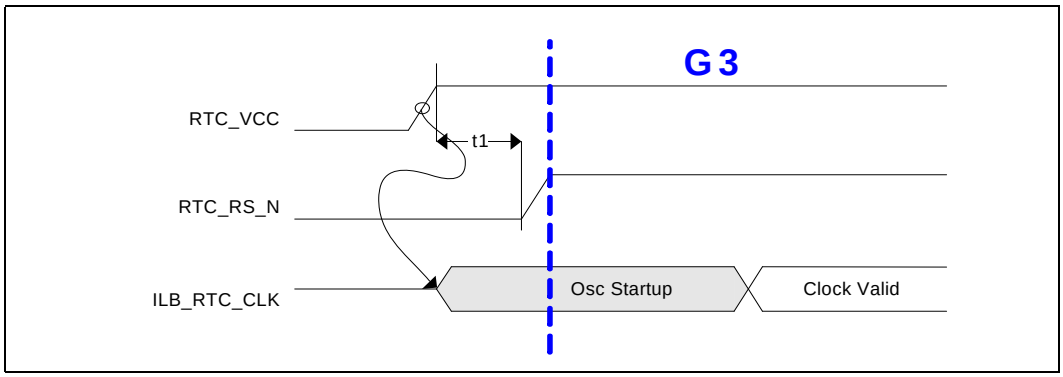


Table 38. RTC Power Well Timing Parameters

Parameter	Description	Min	Max	Units
t1	RTC_VCC to RTC_RST_N de-assertion	9	-	ms

NOTES:
1. This delay is typically created from an RC circuit.
2. The oscillator startup times are component and design specific. A crystal oscillator can take several second to reach a large enough voltage swing. A silicon oscillator can have startups times <10 ms.
3. Pre-silicon estimates

6.2.2 G3 to S4/S5

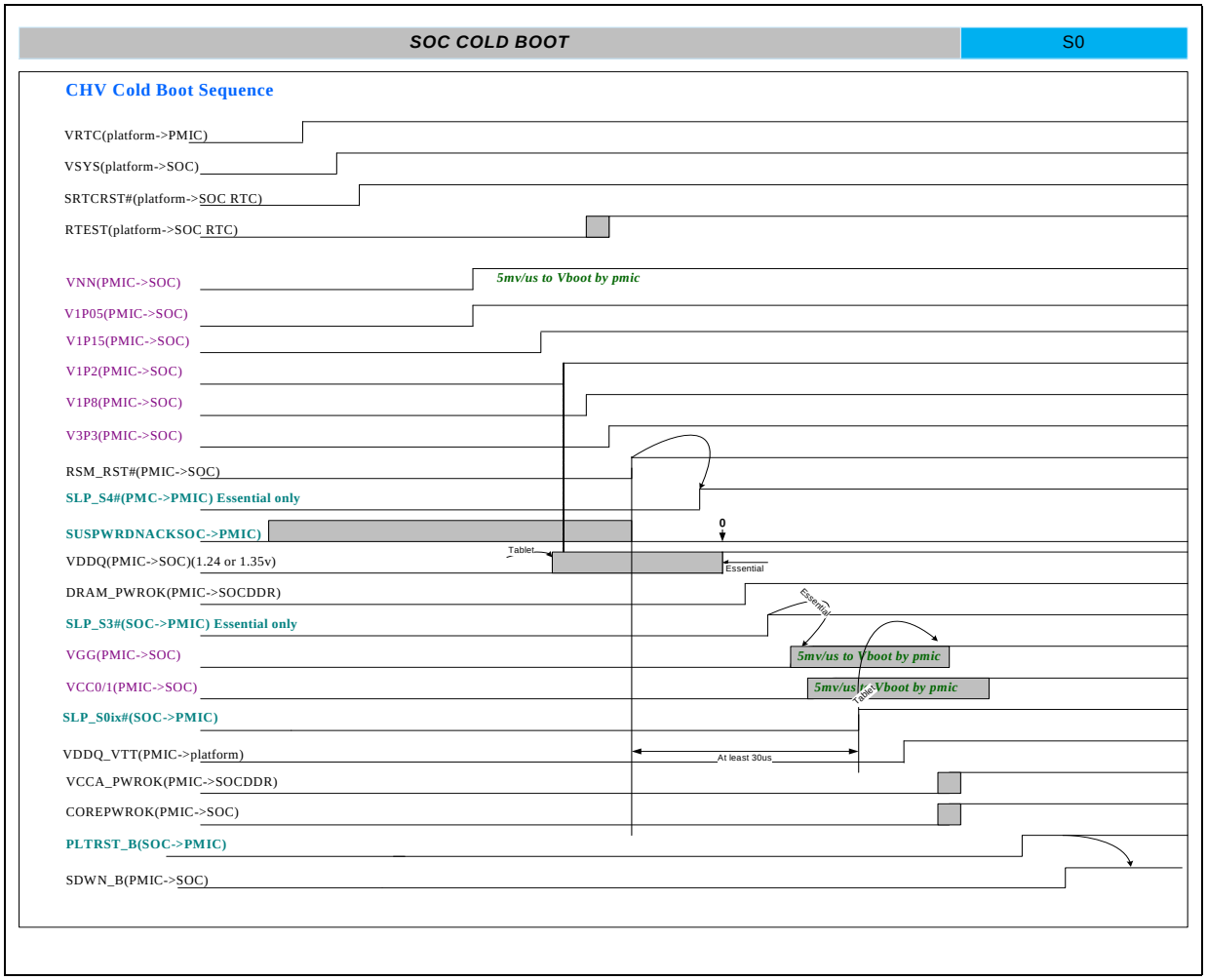
The timings shown in Figure 3 occur when a board event such as AC power or power button is pressed. The following occurs:

1. Suspend well ramp in the order given.
2. The external power management controller de-asserts PMC_RSMRST_N after the suspend rails become stable.
3. PMC_SUSCLK will begin toggling after the de-assertion of PMC_RSMRST_N.
4. The system is now in S4/S5 state. Depending on policy bits, the system either waits for a wake event, or continues to S0 states.

6.2.3 S4/S5 to S0

1. The external power management controller detects an event (i.e., power button) to initiate transition from S4/S5 to S0.
2. VCC, VNN and other S0 core voltage power rails may be enabled after the initiation of the S4/S5 to S0 event. The VCC and VNN voltage rails must be driven to the default values.
3. After the DRAM power rail ramp, the external power management controller drives DRAM_PWROK high.
4. After all of the S0 core voltage power rails are stable, external power management controller drives PMC_CORE_PWROK and VCCA_PWROK to HIGH.
5. The processor de-asserts PMC_PLTRST_N after PMC_CORE_PWROK is stable. The PMC_PLTRST_N is the main platform reset to other components.
6. The processor will begin fetching code from either the PCU-located SPI interface or the LPC interface.

Figure 3. S4/S5 to S0 (Power Up) Sequence



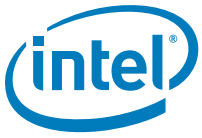
NOTES:.
1. RTC and SUS power rails may come up at the same time if no RTC battery is used.
2. RTC clock should be oscillating, but may not be at 32.768 Khz yet.
3. Wake events show in figure are optional and depending on platform configuration.

6.3 Power Down Sequences

6.3.1 S0 to S4/S5 Sequence

Entry to Sleep states (S4, S5) is initiated by any of the following methods:

- Setting the desired sleep type in PM1_CNT.SLP_TYP and setting PM1_CNT.SLP_EN.
- Detection of an external catastrophic temperature event may cause a transition to G3, if the system is designed to do so.



The following sequence applies to S0-S4/S5 transitions.

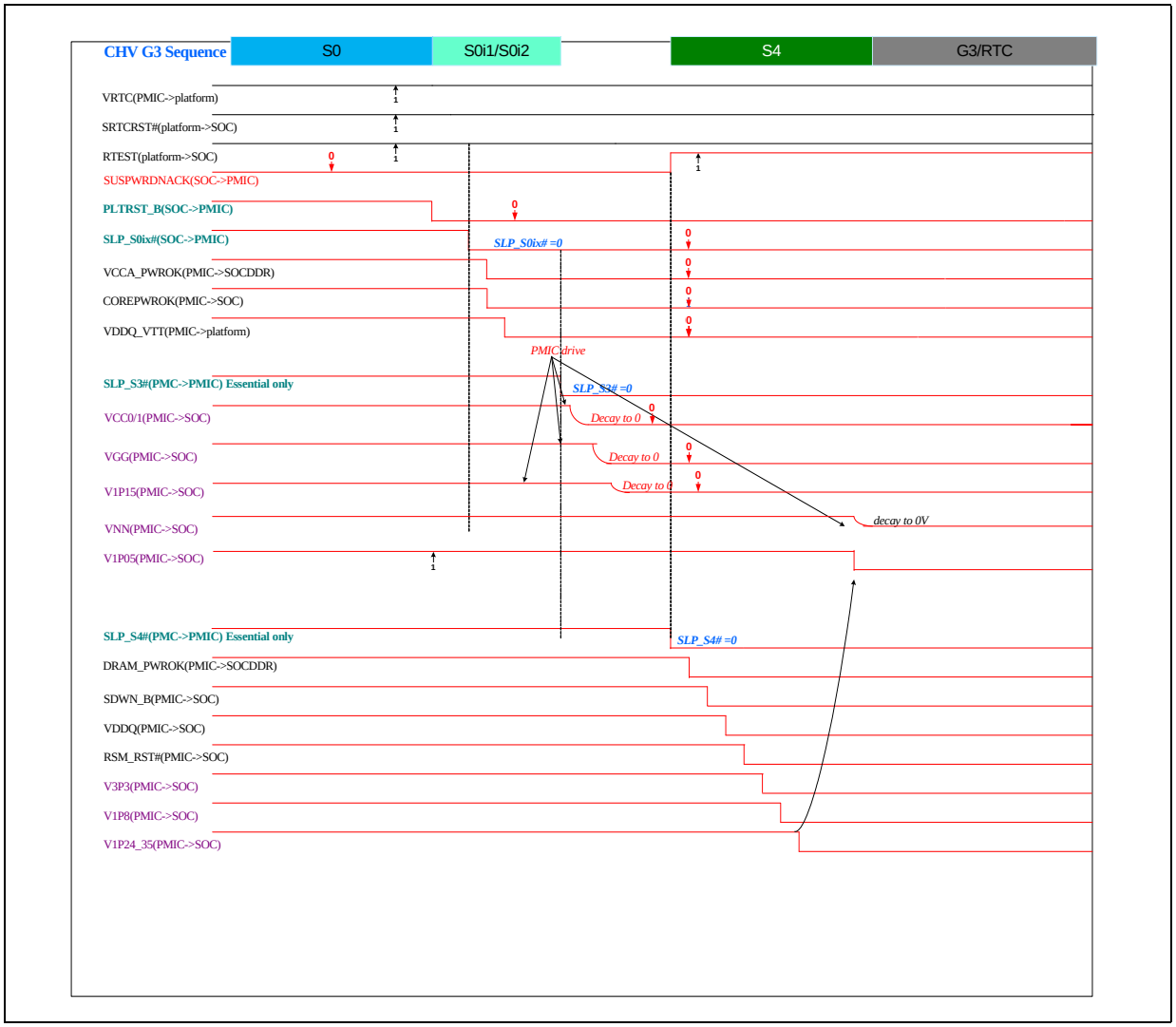
1. The Operating System Power Management (OSPM) will handle the enabling or disabling of interrupt generation after S4 resume. The Operating System Power Management (OSPM) will need to read and clear Wake status information and the processing of the clearing wake status which will include enabling interrupts (both at the core level and platform level).
2. All interrupts in the processor need to be disabled before the S4 sequence is started (and re-enabled on exit). The CPU APIC must be disabled.
3. When the desired sleep state is set in the PM1_CNT.TYP and PM1_CNT.SLP_EN registers, a sleep state request is sent to the PMC.
4. The PMC flushes all the internal buffers to main memory.

The Power Down Sequence is shown in Figure 4 below.

Other Assumptions:

- Entry to a Cx state is mutually exclusive with software-initiated entry to a Sleep state. This is because the processor(s) can only perform one register access at a time. This requirement is enforced by the CPU as well as the OS. The system may hang if it attempts to do a C-state and S-state at the same time.
- The G3 system state cannot be entered via any software mechanism. The G3 state indicates a complete loss of power. In this state, the RTC well may or may not be powered by an external coin cell battery.
- An external Power Management Controller (PMIC/EC) can be used to put the processor in G3 when the S4/S5 state is requested by the SoC. This is done to save power in S4/S5 state. This G3 like state is enabled by removing SUS rails via the SUSPWRDNACK pin. Doing so prevents the use of any of SUS wake events including USB, RTC, and GPIOs including the power button. The external Power Management Controller (or re-application of power) is required to return to S0.

Figure 4. S0 to S4/S5 (Power Down) Sequence



6.3.2 S4/S5 to S0 (Exit Sleep States)

Sleep states (S5) are exited based on Wake events. The Wake events will force the system to a full on state (S0), although some non-critical subsystems might still be powered down and have to be brought back manually. For example, the hard disk may be powered down during a sleep state, and have to be enabled via an I/O pin before it can be used. Upon exit from software-entered Sleep states (i.e., those initiated via the PM1_CNT.SLP_EN bit), the PM1_STS_EN.WAK_STS bit will be set.

To enable Wake Events, the possible causes of wake events (and their restrictions) are shown in Table 39.

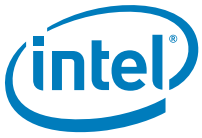


Table 39. S4/S5 to S0 Cause of Wake Events

Cause	Type	How Enabled
RTC Alarm	Internal	Set PM1_STS_EN.RTC_EN register bit
PMC_PWRBTN_N (Power Button)	External	Default enabled as Wake event
GPIO_NORTH And GPIO_SOUTHWEST	External	GPE0a_EN register (after having gone to S5 via PM1_CNT.SLP_EN, but not after a power failure.) Note: GPIOs that are in the core well are not capable of waking the system from sleep states where the core well is not powered.
GPIO_SOUTHEAST	External	Southeast GPIO can (optionally) be used as Wake sources based on GPIO register programming.
Primary PME_N	Internal	GPEOa_EN.PME_B0_EN register bit. This wake status bit includes multiple internal agents: EHCI (USB2)
PMC - Initiated	Internal	No enable bits. The PMC can wake the host independent of other wake events listed, if desired. A bit is provided in PRSTS for reporting this wake event to BIOS. Note that this wake event may be used as a wake trigger on behalf of some other wake source.

6.3.3 Enter S0ix

The S0IX state is entered when the SoC is in a shallow sleep state. This state is entered when the SoC asserts the PMC_SLP_S0IX_N (LOW) pin to the PMIC. VDDQ_VTT and SX rails are turned off. The VCC rail is either turned off by SVID commands (not by PMC_SLP_S0IX_N signal).The VNN rail is set to a voltage set in SVID address 39h.The rest of the VRs remain on but enters into PFM/power save mode.

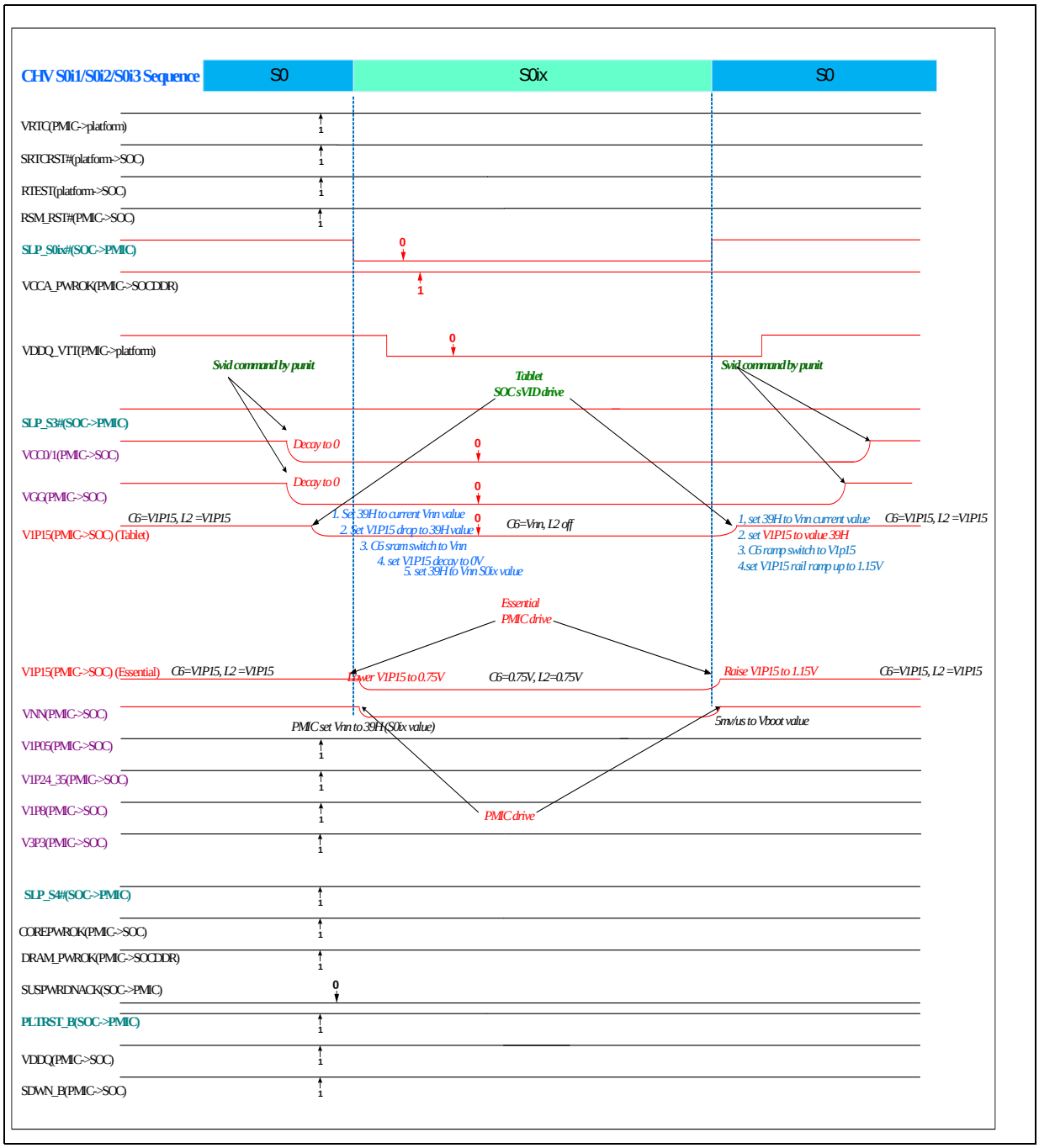
6.3.4 Exit S0ix

The S0IX state is exited when the SoC de-asserts the PMC_SLP_S0IX_Npin (HIGH). VDDQ_VTT and SX rails are turned on. The VCC rail will be turned on by SVID commands (not by PMC_SLP_S0IX_N). The rest of the rails will come out of PFM/power save mode. All SMI/SCI events wake SoC from the S0ix states. The following table lists the addition events that wake the SoC from S0ix states.

Table 40. S0ix Cause of Wake Events

Cause	Type	How Enabled
Any GPIO	External	IO-APIC forwards the interrupt, resulting in S0 (as configured by BIOS). Alternatively use S0ix Wake Register (S0IX_WAKE_EN and S0IX_WAKE_STS) in PMC
LPC CLKRUN	External	Wake from S0i2/3 only when the signal is asserted, moves the SoC to S0i1.
ISH	External	From External Sensors
USB	External	USB Port connected device / host

Figure 5. S0 to S0ix Entry and Exit Sequence



6.3.5 Handling Power Failures

The power failures can occur if the AC power or battery is removed. In this case, when the system was originally in a S0 state, power failure bit (GEN_PMCON1.PWR_FLR) is set after a power failure. Software can clear the bit.

6.4 Reset Behavior

There are several ways to reset the processor.

Table 41. Types of Resets (Sheet 1 of 2)

Trigger	Description	Note
Write of 0Eh to CF9 Register	Write of 0Eh to the CF9 register	TYPE 2: Host Reset with Power Cycle: Cold reset. PMC will lose all the information. All the functionality in SoC gets reset. The host system automatically is powered back up and brought out of reset to S0 state. SoC must not drop this type of reset request if received while the system is in a software-entered S4/5 state.However, SoC is allowed to perform the reset without executing the RESET_WARN protocol in these states. If the system is in S5 due to a reset type #8 event, SoC is allowed to drop this type of reset request.
PMC_RSTBTN_N & CF9h bit 3= 1	User presses the reset button, causing the PMC_RSTBTN_N signal to go active (after the debounce logic)	
PMC_RSTBTN_N & CF9h bit 3= 0	User presses the reset button, causing the PMC_RSTBTN_N signal to go active (after the debounce logic)	TYPE 1:Host Reset with Power Cycle: Warm Reset 1. Host-Only functionality in SoC gets reset 2. Any functionality that needs to remain operational during a host reset must not get reset. 3. PMC does not get reset. 4. RTC remain information. 5. Suspend well remain information 6. S4/S5 drop the warm reset request.
Write of 06h to CF9 Register	Write of 06h to the CF9 register	
TCO watchdog timer	TCO timer reaches zero two times	
S4/S5	The processor is reset when going to S4 or S5 state	TYPE 4: Sx Entry (host stays there) 1. All the Vnn reset by external power Good. Except: 1. PMC remain information. 2. RTC remain information. 3. Suspend well remain information

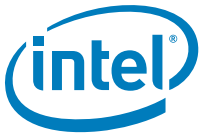


Table 41. Types of Resets (Sheet 2 of 2)

Trigger	Description	Note
Power Failure	PMC_CORE_PWROK signal goes inactive in S0/S1	TYPE 7: Global, Power Cycle Reset: S0->S4/S5->S0 1. All the Vnn reset by external power Good. 2. All power wells that are controlled by the PMC_SLP_S0iX_N pins are turned off. 3. PMC get reset. 4. External Dram-unchanged Except: 1. RTC remain information. 2. Suspend well remain information.
Write of 06h or 0Eh to CF9 Register	CF9h global Reset bit = 1b	
Host Partition Reset Entry Timeout	Host partition reset entry sequence took longer than the allowed timeout value (presumably due to a failure to receive one of the internal or external handshakes)	
Processor Thermal Trip	The internal thermal sensor signals a catastrophic temperature condition – transition to S5 and reset asserts	SOC_G3: Straight-to-S5 (thermal trip->SOC_G3) SOC power cycle: S0->SOC_G3 SOC lost all the info Except: RTC remain info
PMC_PWRBTN_N	10-second press causes transition to S5 (and reset asserts)	TYPE 8: Straight-to-S5 (Host stays there) SOC power cycle: S0->S4->S5 1. All the Vnn reset by external power Good. 2. All power wells that are controlled by the PMC_SLP_S0iX_N pins are turned off. 3. External Dram-unchanged Except: 1. PMC remain information. 2. RTC remain information. 3. Suspend well remain information
PMC_PWRBTN_N Power Button Override		
S4/S5 Entry Timeout	S4, or S5 entry sequence took longer than the allowed timeout value (presumably due to a failure to receive one of the internal or external handshakes)	
PMC Watchdog Timer	Firmware hang and Watchdog Timeout detected in the PMC platform	
CPU Shutdown with Policy to assert PMC_PLTRST_N	Shutdown special cycle from CPU can cause either INIT or Reset Control-style PMC_PLTRST_N	Type 7:Global, Power Cycle Reset (if CF9h Global Reset bit = 1b) Type 2:Host Reset with Power Cycle (if CF9h Register bit 3 = 1b) Type 1:Host Reset without Power Cycle (others setting)

§

7 Thermal Management

The SoC's thermal management system helps in managing the overall thermal profile of the system to prevent overheating and system breakdown. The architecture implements various proven methods of maintaining maximum performance while remaining within the thermal spec. Throttling mechanisms are used to reduce power consumption when thermal limits of the device are exceeded and the system is notified of critical conditions via interrupts or thermal signalling pins. SoC thermal management differs from legacy implementations primarily by replacing dedicated thermal management hardware with firmware.

The thermal management features are:

- Eight digital thermal sensors (DTS)
- Supports a hardware trip point and four programmable trip points based on the temperature indicated by thermal sensors.
- Supports different thermal throttling mechanisms.

7.1 CPU Thermal Management Registers

The description of the control and status registers can be found in the *RS- Cherry Trail SoC BIOS writer's Guide.*

7.2 Thermal Sensors

SoC Sensors are based on DTS (Digital Thermal Sensor) to provide more accurate measure of system thermals.

The SoC has 8 DTS's. DTS provides as wires the current temperature around the real estate it occupies on SoC. These are driven to PM unit, which in turn monitor the temperature from DTS on the SoC.

DTS output are adjusted for silicon variations. For a given temperature the output from DTS is always the same irrespective of silicon.

Table 42. Temperature Reading Based on DTS (Sheet 1 of 2)

DTS Counter Value [8:0]	Temperature Reading (If T_{J-MAX} =90°C)	Temperature Reading (If T_{J-MAX} =100°C)	Temperature Reading (If T_{J-MAX} =110°C)	Temperature Reading (If T_{J-MAX} =100°C) Thermal Read Register [7:0]
127	90°C	100°C	110°C	100°C
137	80°C	90°C	100°C	90°C
147	70°C	80°C	90°C	80°C

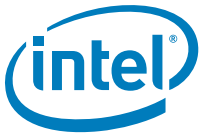


Table 42. Temperature Reading Based on DTS (Sheet 2 of 2)

DTS Counter Value [8:0]	Temperature Reading (If T_{J-MAX} =90°C)	Temperature Reading (If T_{J-MAX} =100°C)	Temperature Reading (If T_{J-MAX} =110°C)	Temperature Reading (If T_{J-MAX} =100°C) Thermal Read Register [7:0]
157	60°C	70°C	80°C	70°C
167	50°C	60°C	70°C	60°C
177	40°C	50°C	60°C	50°C
187	30°C	40°C	50°C	40°C
197	20°C	30°C	40°C	30°C
207	10°C	20°C	30°C	20°C
217	0°C	10°C	20°C	10°C
227	-10°C	0°C	10°C	0°C
237	-20°C	-10°C	0°C	-10°C
247	-30°C	-20°C	-10°C	-20°C
257	-40°C	-30°C	-20°C	-28°C [255]
247	-50°C	-40°C	-30°C	-28°C [255]

Note: DTS encoding of 127 always represents Tjmax. If Tjmax is at 100°C instead of 90°C then the encoding 127 from DTS indicates 100°C, 137 indicates 90°C and so forth.

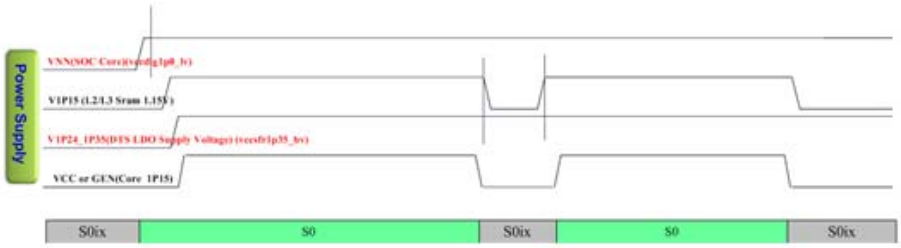
Thermal trip points are of two types:

- **Hard Trip**: The Catastrophic trip points generated by DTS's based on predefined temperature setting defined in fuses.
- **Programmable Trips**: SoC provides four programmable trip settings (Hot, Aux2, Aux1, Aux0) that can be set by firmware/software. Default value for Hot Trip is from Fuses.

7.2.1 DTS Timing

DTS should be enabled only after setting up SoC and system to prevent spurious counts from DTS to trigger thermal events. P-Unit determines when DTS is enabled. The figure below shows the various control signals needed for DTS operations.

Figure 6. DTS Mode of Operation



7.3 Hardware Trips

7.3.1 Catastrophic Trip (THERMTRIP)

Catastrophic trip is generated by DTS whenever the ambient temperature around it reaches (or extends) beyond the max value (indicated by a fuse). Catastrophic trip will not trip unless enabled (DTS are enabled only after HFPLL is locked). Within each DTS Catastrophic trips are flopped to prevent any glitches on Catastrophic signals from affecting the SoC behavior. Catastrophic trips are reset, once set, during power cycles.

Catastrophic trip signals from all DTS in the SoC are combined to generate THERMTRIP function which will in turn shut off all the PLL’**s** and power rails to prevent SoC breakdown. To prevent glitches from triggering shutdown events, Catastrophic trip’s from DTS’s are registered before being sent out.

7.4 SoC Programmable Trips

Programmable trips can be programmed to cause different actions when triggered to reduce temperature of the die.

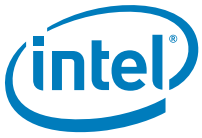
7.4.1 Aux3 Trip

By default, the Aux 3 (Hot Trip) point is set by software/firmware has an option to set these to a different value.

This trip point is enabled by firmware to monitor and control the system temperature while the rest of the system is being set up.

7.4.2 Aux2, Aux1, Aux0 Trip

These are fully programmable trip points for general hardware protection mechanisms. The programmable trips are only active after software/firmware enables the trip.



Note: Unlike Aux3, the Aux[2:0] trip registers are defaulted to zero. To prevent spurious results, software/firmware should program the trip values prior to enabling the trip point.

7.5 Platform Trips

7.5.1 PROCHOT#

The platform components use the signal PROCHOT# to indicate thermal events to SoC. Assertion of the PROCHOT# input will trigger Thermal Monitor 1 or Thermal Monitor 2 throttling mechanisms if they are enabled.

7.5.2 EXTTS

SoC does not support external thermal sensors and the corresponding bits in the P-Unit registers will be reserved for future use if needed.

For SoC, PROCHOT is the only mechanism for a platform component to indicate Thermal events to P-Unit.

7.5.3 sVID

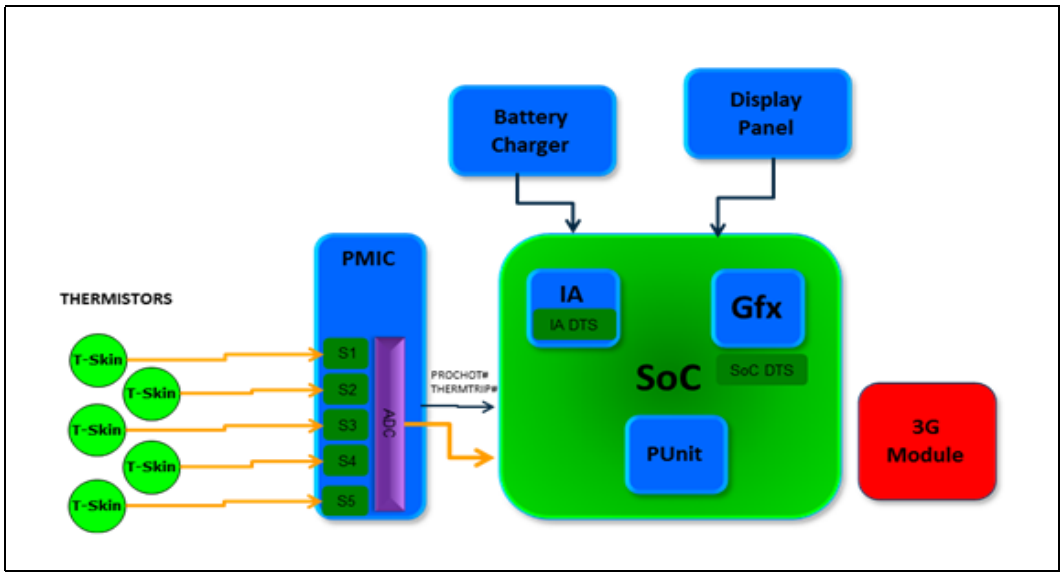
When the Voltage Regulator (VR) reaches it's threshold (VR_Icc_Max, VR_Hot), status bits in sVID are set. sVID sends SVID_Status message to PUnit.

7.6 Dynamic Platform Thermal Framework (DPTF)

SoC is required to support interface for OS level thermal drivers and Intel's DPTF (Dynamic Platform and Thermal Framework) drivers to control thermal management. This interface provides high-level system drivers a mechanism to manage thermal events within the SoC with respect to events outside SoC. These events could potentially be triggered before PM Unit firmware performs active management as DPTF/OS level drivers respond to events on platform outside of SoC.In addition, these interfaces also respond to interrupts from within the SoC.

Platform level thermal management layout is shown in the figure below.

Figure 7. Platform level thermal Management HW Layout

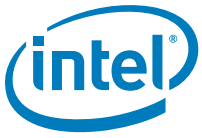


The thermal events happen outside of SoC on platform level are reported as interrupts from PMIC. PMIC monitors a number of catastrophic and critical thermal events, such as PMIC over-temperature, system over-temperature (reported by skin sensors), and battery over-temperature.

7.7 Thermal Status

The firmware captures Thermal Trip events (other than THERMTRIP) in status registers to trigger thermal actions. Associated with each event is a set of programmable actions. For a complete list of refer to the *Cherry trail SoC BIOS Writer's Guide.*

§



8 Power Management

This chapter provides information on the following power management topics:

- ACPI States
- Processor Core
- Integrated Graphics Controller

8.1 Power Management Features

- ACPI System States support (S0, S0i1, S0i2, S0i3, S4, S5)
- Processor Core/Package States support (C0 – C7)
- SoC Graphics Adapter States support D0 – D3.
- Support CPU and GFx Burst
- Dynamic I/O power reductions (disabling sense amps on input buffers, tri-stating output buffers)
- Active power down of Display links

8.2 Power Management States Supported

The Power Management states supported by the processor are described in this section.

8.2.1 System States

Table 43. General Power States for System (Sheet 1 of 2)

States/Sub-states	Legacy Name / Description
G0/S0/C0	**FULL ON:** CPU operating. Individual devices may be shut down to save power. The different CPU operating levels are defined by Cx states.
G0/S0/Cx	**Cx State:** CPU manages C-state itself.
G0/S0i1	**S0i1 State:** Low power platform active state. All DRAM and IOSF traffic are halted. PLL are configured to be off. This state allows MP3 playing using ISH/LPE engine
G0/S0i2	**S0i2 State:** The SoC clocks and oscillators are parked
G0/S0i3	**S0i3 State:** All SoC clocks and oscillators are turned off

Table 43. General Power States for System (Sheet 2 of 2)

States/Sub-states	Legacy Name / Description
G1/S4	**Suspend-To-Disk (STD):** The context of the system is maintained on the disk. All of the power is shut down except power for the logic to resume. The S4 and S5 states are treated the same.
G2/S5	**Soft-Off:** System context is not maintained. All of the power is shut down except power for the logic to restart. A full boot is required to restart. A full boot is required when waking. The S4 and S5 states are treated the same.
G3	**Mechanical OFF.** System content is not maintained. All power shutdown 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, transition will depend on the state just prior to the entry to G3.

Table 46 shows the transitions rules among the various states. Note that transitions among the various states may appear to temporarily transition through intermediate states. These intermediate transitions and states are not listed in the table.

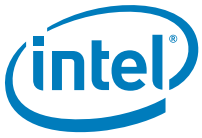
Table 44. Cause of Sx wake events

Cause	How enabled
RTC Alarm	Set RTC_EN bit in PM1_EN Register
Power Button	Always enabled as Wake event from Sx
PMC_SLP_S4_N	None
PMC_BATLOW_N	None
PMC_SUS_STAT_N	None
PMC_SLP_S0IX_N	None
PMC_ACPRESENT	None
PMC_PLTRST_N	None
PMC_SUSCLK[0]	None

The following shows the differences in the sleeping states with regards to the processor's output signals.

Table 45. SoC Sx-States to SLP_S*#

State	S0	S4	S5	Reset w/o Power Cycle	Reset w/ Power Cycle
CPU Executing	In C0	OFF	OFF	No	OFF
PMC_SLP_S4_N	HIGH	LOW	LOW	HIGH	LOW
S0 Power Rails	ON	OFF	OFF	ON	OFF
PMC_PLTRST_N	0	1	1	1	1
PMC_SUS_STAT_N	HIGH	LOW	LOW	HIGH	LOW



Note: The processor treats S4 and S5 requests the same. The processor does not have PMC_SLP_S4_N. PMC_SUS_STAT_N is required to drive low (asserted) even if core well is left on because PMC_SUS_STAT_N also warns of upcoming reset.

Table 46. ACPI PM State Transition Rules

Present State	Transition Trigger	Next State
G0/S0/C0	IA Code MWAIT or LVL Rd	C0/S0/Cx
	PM1_CNT.SLP_EN bit set	G1/Sx or G2/S5 state (specified by PM1_CNT.SLP_TYP)
	Power Button Override	G2/S5
	Mechanical Off/Power Failure	G3
G0/S0/Cx	Cx break events which include: CPU snoop, MSI, Legacy Interrupt, AONT timer	G0/S0/C0
	Power Button Override	G2/S5
	Resume Well Power Failure	G3
G1/S4	Any Enabled Wake Event	G0/S0/C0
	Power button Override	G2/S5
	Resume Well Power Failure	G3
G2/S5	Any Enabled Wake Event	G0/S0/C0
	Resume Well Power Failure	G3
G3	Power Returns	Option to go to S0/C0 (reboot) or G2/S5 (stay off until power button pressed or other enabled wake event) or G1/S4 (if system state was S4 prior to the power failure). Some wake events are preserved through a power failure.

8.2.2 Interface State Combinations

Table 47. G, S and C State Combinations (Sheet 1 of 2)

Global (G) State	Sleep (S) State	Processor Core (C) State	Processor State	System Clocks	Description
G0	S0	C0	Full On	On	Full On
G0	S0	C1/C1E	Auto-Halt	On	Auto-Halt
G0	S0	C6	Deep Power Down	On	Deep Power Down
G0	S0ix	C7	Deep Power Down	On	Deep Power Down
G1	S4	Power off		Off except RTC & internal ring OSC	Suspend to Disk

Table 47. G, S and C State Combinations (Sheet 2 of 2)

Global (G) State	Sleep (S) State	Processor Core (C) State	Processor State	System Clocks	Description
G2	S5	Power off		Off except RTC & internal ring OSC	Soft Off
G3	NA	Power Off		Power off	Hard Off

8.2.3 Integrated Graphics Display States

Table 48. SoC Graphics Adapter State Control

State	Description
D0	Full on, Display active
D3	Power off display

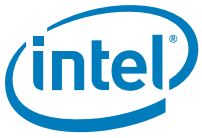
8.2.4 Integrated Memory Controller States

Table 49. Main Memory States

States	Description
Powerup	CKE asserted. Active mode.
Precharge Powerdown	CKE de-asserted (not self-refresh) with all banks closed.
Active Powerdown	CKE de-asserted (not self-refresh) with at least one bank active.
Self-Refresh	CKE de-asserted using device self-refresh

Table 50. D, S and C State Combinations

Graphics Adapter (D) State	Sleep (S) State	(C) State	Description
D0	S0	C0	Full On, Displaying
D0	S0	C1	Auto-Halt, Displaying
D0	S0	C6	Deep Sleep, Display Off
D0	S0ix	C7	Deep Sleep, Display Off
D3	S0/S0ix	Any	Not Displaying
D3	S4		Not Displaying Suspend to disk Core power off



8.3 Processor Core Power Management

While executing code, Enhanced Intel SpeedStep® Technology optimizes the processor's frequency and core 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, lower power C-states have longer entry and exit latencies.

8.3.1 Enhanced Intel SpeedStep® Technology

The following are the key features of Enhanced Intel SpeedStep® Technology:

- Applicable to Processor Core Voltage and Graphic Core Voltage
- Multiple frequency and voltage points for optimal performance and power efficiency. These operating points are known as P-states.
- Frequency selection is software controlled by writing to processor MSRs. The voltage is optimized based on the selected frequency:
  - If the target frequency is higher than the current frequency, Core_VCC is ramped up slowly to an optimized voltage. This voltage is signaled by the SVID signals to the voltage regulator. Once the voltage is established, the PLL locks on to the target frequency.
  - If the target frequency is lower than the current frequency, the PLL locks to the target frequency, then transitions to a lower voltage by signaling the target voltage on the SVID signals.
- The processor controls voltage ramp rates by requesting appropriate ramp rates from an external SVID controller.
- Because there is low transition latency between P-states, a significant number of transitions per second are possible.
- Thermal Monitor mode.
  - Refer to Thermal Management Chapter

8.3.2 Dynamic Cache Sizing*

Dynamic Cache Sizing allows the processor to flush and disable a programmable number of L2 cache ways upon each Deeper Sleep entry under the following condition:

- The C0 timer that tracks continuous residency in the Normal state, has not expired. This timer is cleared during the first entry into Deeper Sleep to allow consecutive Deeper Sleep entries to shrink the L2 cache as needed.
- The predefined L2 shrink threshold is triggered.

Refer to the BIOS Writer's Guide for more details.

8.3.3 Low-Power Idle States

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

8.3.3.1 Clock Control and Low-Power States*

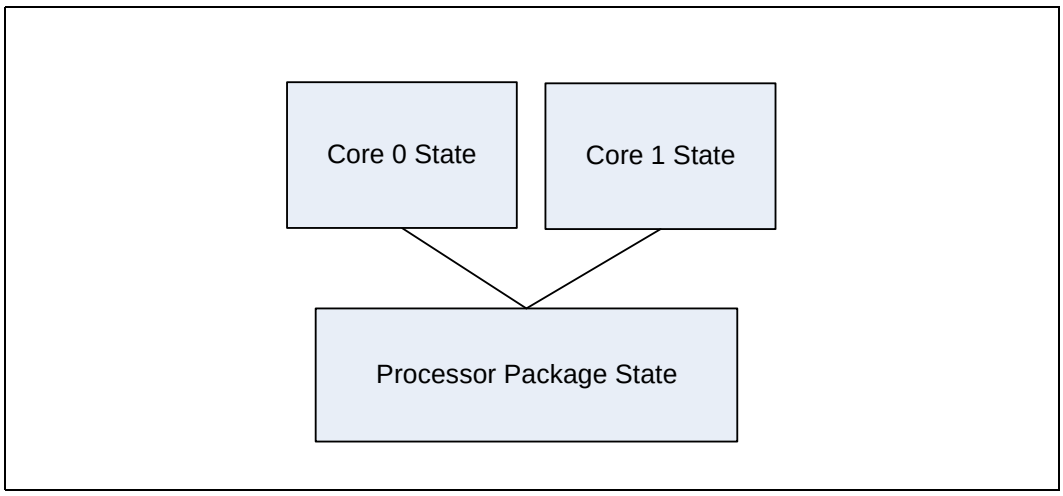
The processor core supports low power states at core level. The central power management logic ensures the entire processor core enters the new common processor core power state. For processor core power states higher than C1, this would be done by initiating a P_LVLx (P_LVL4 & P_LVL6) I/O read to all of the cores. States that require external intervention and typically map back to processor core power states. States for processor core include Normal (C0, C1), and Stop Grant.

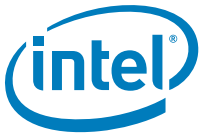
The processor core implements two software interfaces for requesting low power states: MWAIT instruction extensions with sub-state specifies and P_LVLx reads to the ACPI P_BLK register block mapped in the processor core's I/O address space. The P_LVLx I/O reads are converted to equivalent MWAIT C-state requests inside the processor core and do not directly result in I/O reads on the processor core bus. The monitor address does not need to be setup before using the P_LVLx I/O read interface. The sub-state specifications used for each P_LVLx read can be configured in a software programmable MSR by BIOS.

The Cx state ends due to a break event. Based on the break event, the processor returns the system to C0. The following are examples of such break events:

- Any unmasked interrupt goes active
- Any internal event that will cause an NMI or SMI_B
- CPU Pending Break Event (PBE_B)
- MSI

Figure 8. Idle Power Management Breakdown of the Processor Cores





8.3.4 Processor Core C-States Description

Table 51. Processor Core / States Support

State	Description
C0	Active mode, processor executing code
C1	AutoHALT state
C1E	AutoHALT State with lowest frequency and voltage operating point.
C6	Deep Power Down. Prior to entering the Deep Power Down Technology (code named C6) State, The core process will flush its cache and save its core context to a special on die SRAM on a different power plane. Once Deep Power Down Technology (code named C6) sequence has completed. The core processor's voltage is completely shut off.
C7	Execution cores in this state behave similarly to the C6 state. Voltage is removed from the system agent domain

The following state descriptions assume that both threads are in common low power state.

8.3.4.1 Core C0 State

The normal operating state of a core where code is being executed.

8.3.4.2 Core C1 / C1E State

C1/C1E is a low power state entered when a core execute a HLT or MWAIT(C1/C1E) instruction.

A System Management Interrupt (SMI) handler returns execution to either Normal state or the C1/C1E state. See the *Intel® 64 and IA-32 Architecture Software Developer's Manual, Volume 3A/3B: System Programmer's Guide* for more information.

While a core is in C1/C1E state, it processes bus snoops and snoops from other threads. For more information on C1E, see "Section 8.3.5.2, "Package C1/C1E" on page 123".

8.3.4.3 Core C6 State

Individual core can enter the C6 state by initiating a P_LVL3 I/O read or an MWAIT(C6) instruction. Before entering core C6, the core will save its architectural state to a dedicated SRAM.Once complete, a core will have its voltage reduced. During exit, the core is powered on and its architectural state is restored.

8.3.4.4 Core C7 State

Individual core can enter the C7 state by initiating a P_LVL7 I/O read or an MWAIT(C7) instruction. The core C7 state exhibits the same behavior as core C6 state, but in addition gives permission to the internal Power Management logic to enter a package S0ix state if possible.

8.3.4.5 C-State Auto-Demotion*

In general, deeper C-states, such as C6 or C7, have long latencies and higher energy entry/exit costs. The resulting performance and energy penalties become significant when the entry/exit frequency of a deeper C-state is high. Therefore incorrect or inefficient usage of deeper C-states has a negative impact on battery life. In order to increase residency and improve battery life in deeper C-states, the processor supports C-state auto-demotion.

This is the C-State auto-demotion option:

- C7/C6 to C1

The decision to demote a core from C7/C6 to C1 is based on each core's immediate residency history. Upon each core C7/C6 request, the core C-state is demoted to C1 until a sufficient amount of residency has been established. At that point, a core is allowed to go into C6 or C7.

This feature is disabled by default. BIOS must enable it in the PMG_CST_CONFIG_CONTROL register. The auto-demotion policy is also configured by this register. See the BIOS Writer's Guide for more details.

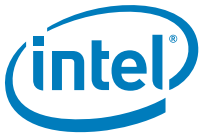
8.3.5 Package C-States*

The processor supports C0, C1/C1E,C6 and C7 power 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:

- Package C-state request is determined by the lowest numerical core C-state amongst all cores.
- A package C-state is automatically resolved by the processor depending on the core idle power states and the status of the platform components.
- Each 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.

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 core break event is received, the target core is activated and the break event message is forwarded to the target core.



— If the break event is not masked, the target core enters the 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 52. Coordination of Core / Module Power States at the Package Level

Package C-State		Core / Module 1				
		C0	C1	C6NS	C6FS	C7
Core / Module 0	C0	C0	C1[1]	C0	C0	C0
	C1	C0	C1[1]	C1	C1[1]	C1[1]
	C6NS	C0	C1[1]	C6C	C6	C6
	C6FS	C0	C1[1]	C6C	C6	C6
	C7	C0	C1[1]	C6C	C6	C7

NOTE:
1. If enabled, the package C-state will be C1E if all actives cores have resolved a core C1 state or higher.
2. C6C is C6-Conditional where the l2 cache is still powered.
3. 2 Cores of the SoC will make up one module.

8.3.5.1 Package C0

The normal operating state for the processor. The processor remains in the normal state when at least one of its cores is in the C0 state or when the platform has not granted permission to the processor to go into a low power state. Individual cores may be in lower power idle states while the package is in C0.

8.3.5.2 Package C1 / C1E

No additional power reduction actions are taken in the package C1 state. However, if the C1E sub-state is enabled, the processor automatically transitions to the lowest supported core clock frequency, followed by a reduction in voltage.

The package enters the C1 low power state when:

- At least one core is in the C1 state.
- The other cores are in a C1 or lower power state.

The package enters the C1E state when:

- All cores have directly requested C1E via MWAIT(C1) with a C1E sub-state hint.

- All cores are in a power state lower that C1/C1E but the package low power state is limited to C1/C1E via the PMG_CST_CONFIG_CONTROL MSR.
- All cores have requested C1 using HLT or MWAIT(C1) and C1E auto-promotion is enabled in IA32_MISC_ENABLES.

No notification to the system occurs upon entry to C1/C1E.

8.3.5.3 Package C6 State

A processor enters the package C6 low power state when:

- At least one core is in the C6 state.
- The other cores are in a C6 or lower power state, and the processor has been granted permission by the platform.
- The platform has not granted a request to a package C7 state but has allowed a package C6 state.

In package C6 state, all cores have saved their architectural state and have had their core voltages reduced to zero volts.

8.3.5.4 Package C7 State

A processor enters the package C7 low power state when all cores are in the C7 state. In package C7, the processor will take action to remove power from portions of the system agent.

Core break events are handled the same way as in package C6.

8.3.6 Graphics, Video and Display Power Management

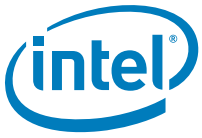
8.3.6.1 Graphics and video decoder C-State

GFX C-State (GC6) are designed to optimize the average power to the graphics and video decoder engines during times of idleness. GFX C-state is entered when the graphics engine, has no workload being currently worked on and no outstanding graphics memory transactions. When the idleness condition is met, the processor will power gate the Graphics and video decoder engines.

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

The Intel DPST technique achieves backlight power savings while maintaining visual experience. This is accomplished by adaptively enhancing the displayed image while decreasing the backlight brightness simultaneously. The goal of this technique is to provide equivalent end-user image quality at a decreased backlight 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 software algorithm determines that



enough brightness, contrast, or color change has occurred to the displaying images that the image enhancement and backlight control needs to be altered.)

2. Intel DPST subsystem applies an image-specific enhancement to increase image contrast, brightness, and other attributes.
3. A corresponding decrease to the backlight brightness is applied simultaneously to produce an image with similar user-perceived quality (such as brightness) as the original image. Intel DPST 6.0 has improved the software algorithms and has minor hardware changes to better handle backlight phase-in and ensures the documented and validated method to interrupt hardware phase-in.

8.3.6.3 Intel® Automatic Display Brightness

The Intel Automatic Display Brightness feature dynamically adjusts the backlight brightness based upon the current ambient light environment. This feature requires an additional sensor to be on the panel front. The sensor receives the changing ambient light conditions and sends the interrupts to the Intel Graphics driver. As per the change in Lux, (current ambient light illuminance), the new backlight setting can be adjusted through BLC. The converse applies for a brightly lit environment. Intel Automatic Display Brightness increases the back light setting.

8.3.6.4 Intel® Seamless Display Refresh Rate Switching Technology (Intel® SDRRS Technology)

When a Local Flat Panel (LFP) supports multiple refresh rates, the Intel® Display Refresh Rate Switching power conservation feature can be enabled. The higher refresh rate will be used when on plugged in power or when the end user has not selected/ enabled this feature. The graphics software will automatically switch to a lower refresh rate for maximum battery life when the design application is on battery power and when the user has selected/enabled this feature.

There are two distinct implementations of Intel SDRRS—static and seamless. The static Intel SDRRS method uses a mode change to assign the new refresh rate. The seamless Intel SDRRS method is able to accomplish the refresh rate assignment without a mode change and therefore does not experience some of the visual artifacts associated with the mode change (SetMode) method.

8.4 Memory Power Management

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

8.4.1 Disabling Unused System Memory Outputs

Any system memory (SM) interface signal that goes to a memory module connector in which it is not connected to any actual memory devices (such as DIMM 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 un-terminated transmission lines.

When a given rank is not populated, the corresponding chip select and CKE signals are not driven.

SCKE tri-state should be enabled by BIOS where appropriate, since at reset all rows must be assumed to be populated.

8.4.2 DRAM Power Management and Initialization

The processor implements extensive support for power management on the SDRAM interface. There are four SDRAM operations associated with the Clock Enable (CKE) signals, which the SDRAM controller supports. The processor drives four CKE pins to perform these operations.

8.4.2.1 Initialization Role of CKE*

During power-up, CKE is the only input to the SDRAM that is recognized (other than the DDR3 reset pin) once power is applied. It must be driven LOW by the DDR controller to make sure the SDRAM components float DQ and DQS during power- up.

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

8.4.2.2 Conditional Self-Refresh

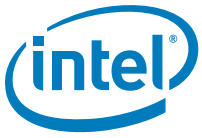
Intel Rapid Memory Power Management (Intel RMPM) conditionally places memory into self-refresh in the package low-power states. RMPM functionality depends on graphics/ display state (relevant only when internal graphics is being used), as well as memory traffic patterns generated by other connected I/O devices.

When entering the Suspend-to-RAM (STR) state, the processor core flushes pending cycles and then places all SDRAM ranks into self refresh. In STR, the CKE signals remain LOW so the SDRAM devices perform self-refresh.

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

8.4.2.3 Dynamic Power Down Operation

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 core controller can be configured to put the devices in active power down (CKE deassertion with open pages) or precharge power down (CKE deassertion with all



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

If dynamic power-down is enabled, all ranks are powered up before doing a refresh cycle and all ranks are powered down at the end of refresh.

8.4.2.4 DRAM I/O Power Management*

Unused signals should be disabled to save power and reduce electromagnetic interference. This includes all signals associated with an unused memory channel. Clocks can be controlled on a per SO-DIMM basis. Exceptions are made for per SO-DIMM control signals such as CS#, CKE, and ODT for unpopulated SO-DIMM slots.

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

§

9 System Memory Controller

The system memory controller supports DDR3L-RS/LPDDR3 protocol with up to two 64-bit wide dual rank channels at data rates up to 1600 MT/s with ECC is also available on a single DDR3L-RS channel.

Note: Cherry Trail CO-POP Package will support PoP (Package on Package) memory devices.

9.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

9.1.1 DDR3L-RS Interface Signals

Table 53. Memory Channel 0 DDR3L-RS Signals (Sheet 1 of 2)

Signal Name	Direction Type	Description
DDR3_M0_CK[1,0]_P **DDR3_M0_CK[1,0]_N**	O DDR3	**Clock PAD:** (1 pair per Rank) Driven by PHY to DRAM.
DDR3_M0_CS[1,0]_N	O DDR3	**Chip Select:** (1 per Rank). Driven by PHY to DRAM.
DDR3_M0_CKE[3,0]	O DDR3	**Clock Enable:** (power management) Driven by PHY to DRAM.
DDR3_M0_MA[15:0]	O DDR3	**Memory Address:** Driven by PHY to DRAM.
DDR3_M0_BS[2:0]	O DDR3	**Bank Select:** Driven by PHY to DRAM.
DDR3_M0_RAS_N	O DDR3	**Row Address Select:** Used with DDR3_M0_CAS# and DDR3_M0_WE# (along with DDR3_M0_CS#) to define the DRAM Commands
DDR3_M0_CAS_N	O DDR3	**Column Address Select:** Used with DDR3_M0_RAS# and DDR3_M0_WE# (along with DDR3_M0_CS#) to define the SRAM Commands
DDR3_M0_WE_N	O DDR3	**Write Enable Control Signal:** Used with DDR3_M0_WE# and DDR3_M0_CAS# (along with control signal, DDR3_M0_CS#) to define the DRAM Commands.

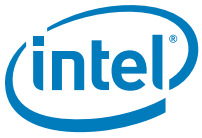


Table 53. Memory Channel 0 DDR3L-RS Signals (Sheet 2 of 2)

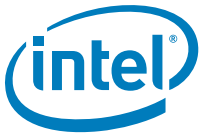
Signal Name	Direction Type	Description
DDR3_M0_DQ[63:0]	I/O DDR3	**Data Lines:** Bidirectional signals between DRAM/ PHY
DDR3_M0_DM[7:0]	O DDR3	**Data Mask:** DM is an output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of DQS.
DDR3_M0_DQS[7:0]_P **DDR3_M0_DQS[7:0]_N**	I/O DDR3	**Data Strobes:** The data is captured at the crossing point of each 'P' and its compliment 'N' during read and write transactions. For reads, the strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window.
DDR3_M0_ODT[1,0]	O DDR3	**On Die Termination:** ODT signal going to DRAM in order to turn ON the DRAM ODT during Write.
DDR3_M0_RCOMPD	I DDR	**Resistor Compensation:** This signal needs to be terminated to VSS on board.This signal is driven from external clock source.
DDR3_M0_OCAVREF	I DDR	**Reference Voltage:** DDR3 CA interface Reference Voltage
DDR3_M0_ODQVREF	I DDR	**Reference Voltage:** DDR3 DQ interface Reference Voltage
DDR3_CORE_PWROK	I DDR	**Core Power OK:** This signal indicates the status of the DRAM Core power supply (power on in S0).Active high signal indicates that DDR PHY voltage(1.5v) is good.
DDR3_VDD_S4_PWROK	I DDR	**VDD Power OK:** Asserted once the VRM is settled.
DDR3_M0_DRAMRST_N	O	**DRAM Reset:** This signal is used to reset DRAM devices.

Table 54. Memory Channel 1 DDR3L-RS Signals (Sheet 1 of 2)

Signal Name	Direction Type	Description
DDR3_M1_CK[1,0]_P **DDR3_M1_CK[1,0]_N**	O DDR3	**Clock PAD:** (1 pair per Rank) Driven by PHY to DRAM.
DDR3_M1_CS[1,0]_N	O DDR3	**Chip Select:** (1 per Rank). Driven by PHY to DRAM.
DDR3_M1_CKE[3,0]	O DDR3	**Clock Enable:** (power management) Driven by PHY to DRAM.
DDR3_M1_MA[15:0]	O DDR3	**Memory Address:** Driven by PHY to DRAM.
DDR3_M1_BS[2:0]	O DDR3	**Bank Select:** Driven by PHY to DRAM.

Table 54. Memory Channel 1 DDR3L-RS Signals (Sheet 2 of 2)

Signal Name	Direction Type	Description
DDR3_M1_RAS_N	O DDR3	**Row Address Select:** Used with DDR3_M0_CAS# and DDR3_M0_WE# (along with DDR3_M0_CS#) to define the DRAM Commands
DDR3_M1_CAS_N	O DDR3	**Column Address Select:** Used with DDR3_M0_RAS# and DDR3_M0_WE# (along with DDR3_M0_CS#) to define the SRAM Commands
DDR3_M1_WE_N	O DDR3	**Write Enable Control Signal:** Used with DDR3_M0_WE# and DDR3_M0_CAS# (along with control signal, DDR3_M0_CS#) to define the DRAM Commands.
DDR3_M1_DQ[63:0]	I/O DDR3	**Data Lines:** Bidirectional signals between DRAM/PHY
DDR3_M1_DM[7:0]	O DDR3	**Data Mask:** DM is an output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of DQS.
DDR3_M1_DQS[7:0]_P **DDR3_M1_DQS[7:0]_N**	I/O DDR3	**Data Strobes:** The data is captured at the crossing point of DDR3_M1_DQSP[7:0] and its compliment 'N' during read and write transactions. For reads, the strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window.
DDR3_M1_ODT[1,0]	O DDR3	**On Die Termination:** ODT signal going to DRAM in order to turn ON the DRAM ODT during Write.
DDR3_M1_DRAMRST_N	O	**Reset DRAM:** This signal can be used to reset DRAM devices.



9.1.2 LPDDR3 Interface Signals

Table 55. Memory Channel 0 LPDDR3 Signals

Signal Name	Direction Type	Description
LPDDR3_M0_CK_P_A/B **LPDDR3_M0_CK_N_A/B**	O DDR3	**SDRAM and inverted Differential Clock:** (1 pair per Rank) The differential clock pair is used to latch the command into DRAM. Each pair corresponds to one rank on DRAM side.
LPDDR3_M0_CS[1,0]_N	O DDR3	**Chip Select:** (1 per Rank). Used to qualify the command on the command bus for a particular rank.
LPDDR3_M0_CKE[1,0]_A/B	O DDR3	**Clock Enable:** (power management) It is used during DRAM power up/power down and Self refresh. *Note: LPDDR3L uses only LPDDR3_M0_CKE[2,0]. LPDDR3_M0_CKE[1,3] are not being used for LPDDR3L.*
LPDDR3_M0_CA[9:0]	O DDR3	**Memory Address:** Memory address bus for writing data to memory and reading data from memory. These signals follow common clock protocol w.r.t. LPDDR3_M0_CKN, LPDDR3_M0_CKP pairs
LPDDR3_M0_DQ[31:0]_A/B	I/O DDR3	**Data Lines:** Data signal interface to the DRAM data bus
LPDDR3_M0_DM[3:0]_A/B	O DDR3	**Data Mask:** DM is an output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of DQS.
LPDDR3_M0_DQS[3:0]_P_A/B **LPDDR3_M0_DQS[3:0]_N_A/B**	I/O DDR3	**Data Strobes:** The data is captured at the crossing point of each 'P' and its compliment 'N' during read and write transactions. For reads, the strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window.
LPDDR3_M0_ODT_A/B	O DDR3	**On Die Termination:** ODT signal going to DRAM in order to turn ON the DRAM ODT during Write.
LPDDR3_M0_RCOMPD	I DDR	**Resistor Compensation:** This signal needs to be terminated to VSS on board. This signal is driven from external clock source.
LPDDR3_M0_OCAVREF	I DDR	**Reference Voltage:** LPLPDDR3 CA interface Reference Voltage
LPDDR3_M0_ODQVREF	I DDR	**Reference Voltage:** LPLPDDR3 DQ interface Reference Voltage

Table 56. Memory Channel 1 LPDDR3L-RS Signals

Signal Name	Direction Type	Description
LPDDR3_M1_CK_P_A/B **LPDDR3_M1_CK_N_A/B**	O DDR3	**SDRAM and inverted Differential Clock:** (1 pair per Rank) The differential clock pair is used to latch the command into DRAM. Each pair corresponds to one rank on DRAM side.
LPDDR3_M1_CS[1,0]_N	O DDR3	**Chip Select:** (1 per Rank). Used to qualify the command on the command bus for a particular rank.
LPDDR3_M1_CKE[1,0]_A/B	O DDR3	**Clock Enable:** (power management) It is used during DRAM power up/power down and Self refresh. *Note: LPDDR3L uses only LPDDR3_M1_CKE[0,2]. LPDDR3_M1_CKE[1,3] are not being used for LPDDR3L.*
LPDDR3_M1_CA[9:0]	O DDR3	**Memory Address:** Memory address bus for writing data to memory and reading data from memory. These signals follow common clock protocol relative to LPDDR3_M1_CKN, LPDDR3_M1_CKP pairs
LPDDR3_M1_DQ[31:0]_A/B	I/O DDR3	**Data Lines:** Data signal interface to the DRAM data bus.
LPDDR3_M1_DM[3:0]_A/B	O DDR3	**Data Mask:** DM is an output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of DQS.
LPDDR3_M1_DQS[3:0]_P_A/B **LPDDR3_M1_DQS[3:0]_N_A/B**	I/O DDR3	**Data Strobes:** The data is captured at the crossing point of LPDDR3_M1_DQSP[7:0] and its compliment 'N' during read and write transactions. For reads, the strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window.
LPDDR3_M1_ODT_A/B	O DDR3	**On Die Termination:** ODT signal going to DRAM in order to turn ON the DRAM ODT during Write.
LPDDR3_M1_OCAVREF	I DDR	**Reference Voltage:** LPDDR3 CA interface Reference Voltage
LPDDR3_M1_ODQVREF	I DDR	**Reference Voltage:** LPDDR3 DQ interface Reference Voltage

9.1.3 ECC Support

The system memory controller supports ECC. When ECC is enabled, only Memory Channel 0 will be active. Memory Channel 1 will be disabled and used for the ECC data pins. The table below shows the details on the muxing relationship between the ECC Signals and the Memory Channel 1 signals.

Note: ECC SO-DIMMs are not backwards compatible with non-ECC SO-DIMMs.

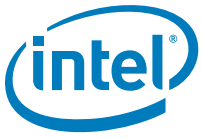


Table 58. ECC Signals

Signal Name	Direction Type	Description
DDR3_M0_ECC_DQ[7:0]	I/O DDR3	**ECC Check Data Bits** *These are muxed with channel 1.*
DDR3_M0_ECC_DM	O DDR3	**ECC Data Mask:** DM is an optional output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of ECC_DQS. *This signal is muxed with channel 1 and may not be needed.*
DDR3_M0_ECC_DQSP **DDR3_M0_ECC_DQSN**	I/O DDR3	**ECC Data Strobes:** The data is captured at the crossing point the 'P' and its compliment 'N' during read and write transactions. For reads, the strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window. *These are muxed with channel 1.*

9.2 Features

9.2.1 System Memory Technology Supported

The system memory controller supports the following DDR3L-RS/LPDDR3 Data Transfer Rates, SO-DIMM Modules and DRAM Device Technologies:

- DDR3L-RS/LPDDR3 Data Transfer Rates: 1600MT/s (12.8 GB/s per channel)
- LPDDR3 (1.2V DRAM VDDQ)
- DDR3L-RS (1.35V DRAM interface I/Os)
- DDR3L-RS DRAM Device Technology
  - Standard 4Gb technologies and addressing.
  - Read latency 5, 6, 7, 8, 9, 10, 11, 12, 13
  - Write latency 3, 4, 5, 6, 7, 8
- LPDDR3 DRAM Device Technology
  - x64, 253 ball LPDDR3 DRAM package.
  - 8 GB (1 rank per channel), 16 GB (2 rank per channel) package density.
  - Standard 2 GB, 4 GB and 8 GB, x32b DRAM technologies and addressing.
  - Read latency 5, 6, 7, 8, 9, 10, 11, 12, 13
  - Write latency 3, 4, 5, 6, 7, 8
- Support Trunk Clock Gating
- ECC support for 64-bit data bus on DDR3L-RS single channel
- Support early SR exit

- Support slow power down
- Support CA tri-state when not driving a valid command.

9.2.2 CO-POP package Configuration

The Cherry Trail CO-POP will support PoP (Package on Package) memory devices.

The LPDDR3 memory device package will be attached directly on top of the processor package & hence this is called a package-on-package [POP] configuration between the memory controller & the LPDDR3 system memory.

To aid seamless connection between the memory controller and the LPDDR3 device; the memory controller package balls that communicate with LPDDR3 are brought out on all the 4 top sides of the package. The interconnect route from the memory controller to LPDDR3 is essentially the transmission line on the package. No board interconnect routing exists for the memory subsystem and hence the usual signal integrity distortion on the memory channels is substantially reduced.

Figure below shows different components in a POP topology. Intel will supply SOC with the PKG interposer on which LPDDR3 modules can be assembled.

Figure 9. Co-POP Overview Block Diagram

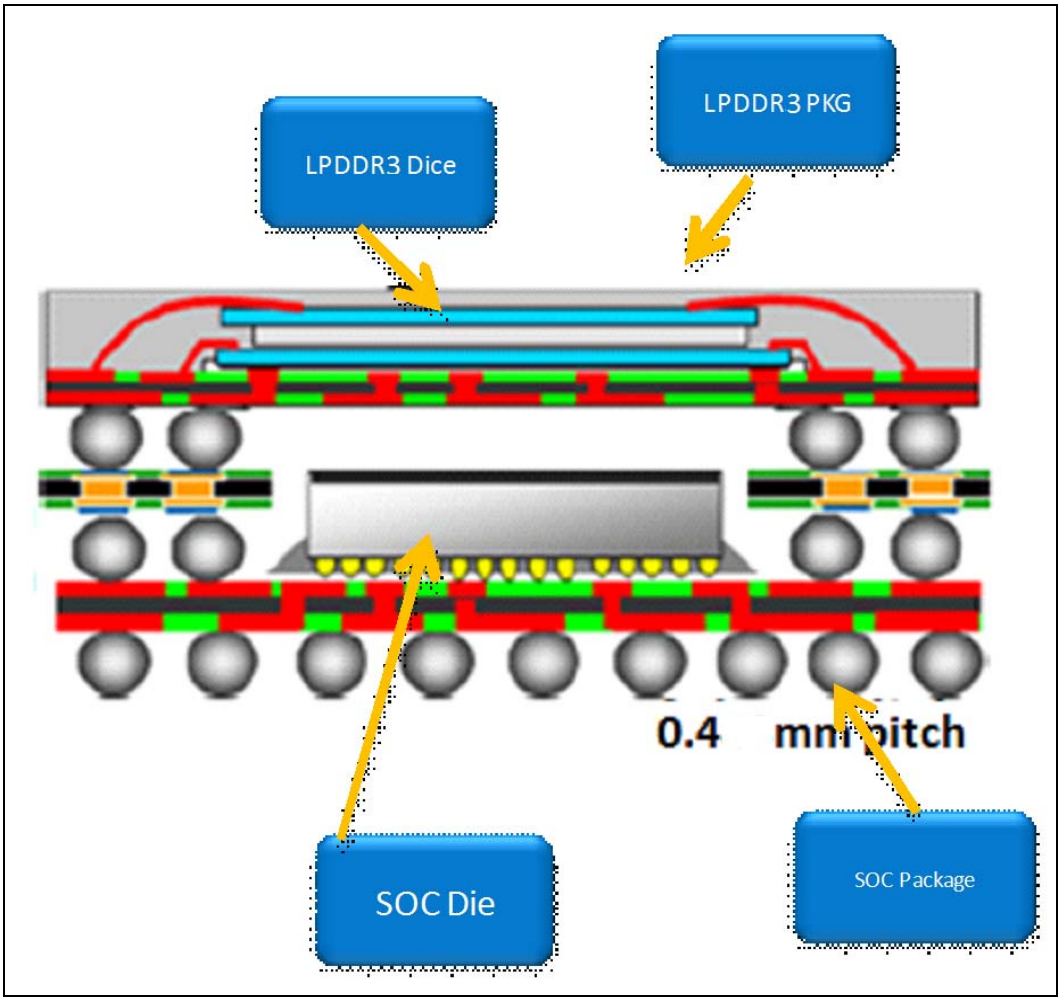


Table 59. Supported LPDDR3 DRAM Devices

DRAM Density	Data Width	Banks	Bank Address	Row Address	Column Address	Page Size
2Gb	x32	8	BA[2:0]	A[13:0]	A[8:0]	2KB
4Gb	x32	8	BA[2:0]	A[13:0]	A[9:0]	4KB
8Gb	x32	8	BA[2:0]	A[14:0]	A[9:0]	4KB

Table 60. Supported DDR3L-RS DRAM Devices

DRAM Density	Data Width	Banks	Bank Address	Row Address	Column Address	Page Size
4Gb	x8	8	BA[2:0]	A[15:0]	A[9:0]	1KB
4Gb	x16	8	BA[2:0]	A[14:0]	A[9:0]	2KB

Table 61. Supported LPDDR3 Memory Size Per Rank

Memory Size/ Rank	DRAM Chips/ Rank	DRAM Chip Density	DRAM Chip Data Width	Page Size @ 64-bit Data Bus
512MB	2	2Gb	x16	4KB = 2KB * 2 chips
1GB	2	4Gb	x16	8KB = 4KB * 2 chips
2GB	2	8Gb	x16	8KB = 4KB * 2 chips

Table 62. Supported DDR3L-RS Memory Size Per Rank

Memory Size/ Rank	DRAM Chips/ Rank	DRAM Chip Density	DRAM Chip Data Width	Page Size @ 64-bit Data Bus
4GB	8	4Gb	x8	8KB = 1KB * 8 chips
2GB	4	4Gb	x16	8KB = 2KB * 4 chips

9.3 Register Map

For more information on System Memory Controller registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§

10 Graphics, Video and Display

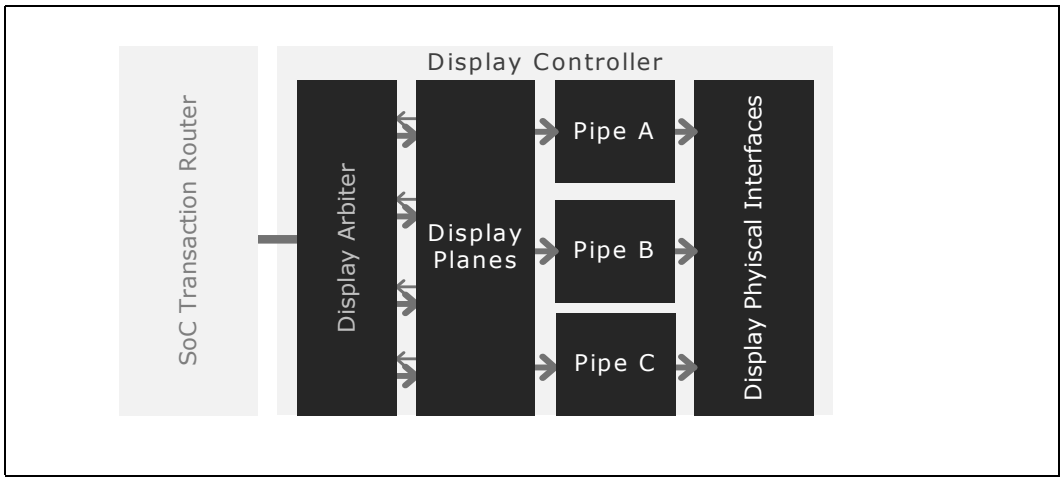
This chapter provides an overview of Graphics, Video and Display features of the SoC.

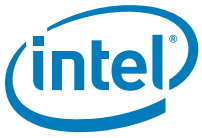
10.1 Features

The key features of the individual blocks are as follows:

- Refreshed eight generation Intel graphics core with sixteen Execution Units (EUs)
  - 3D graphics hardware acceleration including support for DirectX*11.1, OpenGL 4.3, OGL ES 3.0, OpenCL 1.2.
  - Video decode hardware acceleration including support for HEVC (H.265), H.264 @5.1, SVC, VP8 1080p, VP8 4K, VP8 stereo, MPEG4 ASP/SP, AVS, BD2.4, H.263 and VP9 formats
  - Video encode hardware acceleration including support for H.264@ 4.2, VP8 1080p, VP8 4K, VP8 stereo, SVC, AVS, H.263 and VP9 formats.
  - Display controller, incorporating the display planes, pipes and physical interfaces
  - Four planes available per pipe - 1x Primary, 2x Video Sprite & 1x Cursor
  - Three multi-purpose Digital Display Interface (DDI) PHYs implementing HDMI, DVI, DisplayPort (DP) or Embedded DisplayPort (eDP) support
  - Two dedicated digital Display Serial Interface PHYs implementing MIPI-DSI support

10.2 SoC Graphics Display





The Processor Graphics controller display pipe can be broken down into three components:

- Display Planes
- Display Pipes
- Display Physical Interfaces

A display plane is a single displayed surface in memory and contains one image (desktop, cursor, overlay). It is the portion of the display hardware logic that defines the format and location of a rectangular region of memory that can be displayed on a display output device and delivers that data to a display pipe. This is clocked by the Core Display Clock.

10.2.1 Primary Planes A,B and C

Planes A, B and C are the main display planes and are associated with Pipes A, B and C respectively. Each plane supports per-pixel alpha blending.

10.2.2 Video Sprite Planes A, B, C, D, E and F

Video Sprite Planes A, B, C, D, E and F are planes optimized for video decode.

- Pipe A – Primary planeA, VSpriteA, VSpriteB, CusrorA
- Pipe B – Primary planeB, VSpriteC, VSpriteD, CursorB
- Pipe C – Primary planeC, VSpriteE, VSpriteF, CursorC

10.2.3 Cursors A, B and C

Cursors A, B and C are small, fixed-sized planes dedicated for mouse cursor acceleration, and are associated with Planes A, B and C respectively.

10.3 Display Pipes

The display pipe blends and synchronizes pixel data received from one or more display planes and adds the timing of the display output device upon which the image is displayed.

The display pipes A, B and C operate independently of each other at the rate of one pixel per clock. They can attach to any of the display interfaces.

10.4 Display Physical Interfaces

The display physical interfaces consist of output logic and pins that transmit the display data to the associated encoding logic and send the data to the display device. These interfaces are digital (MIPI-DSI, DisplayPort*, Embedded DisplayPort*, DVI and HDMI*) interfaces.

Figure 10. Display Pipe to Port Mapping [MSP-T4]

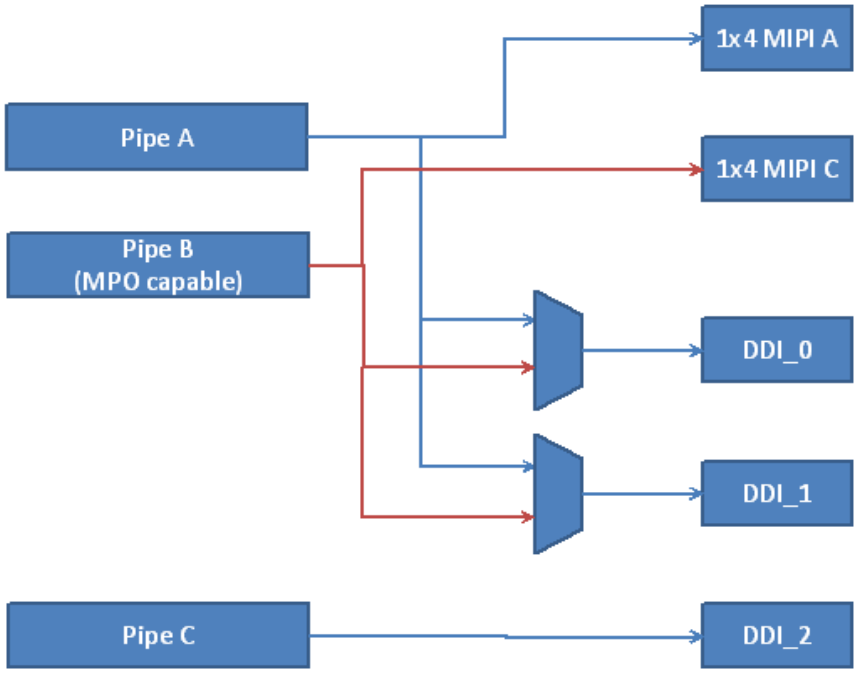
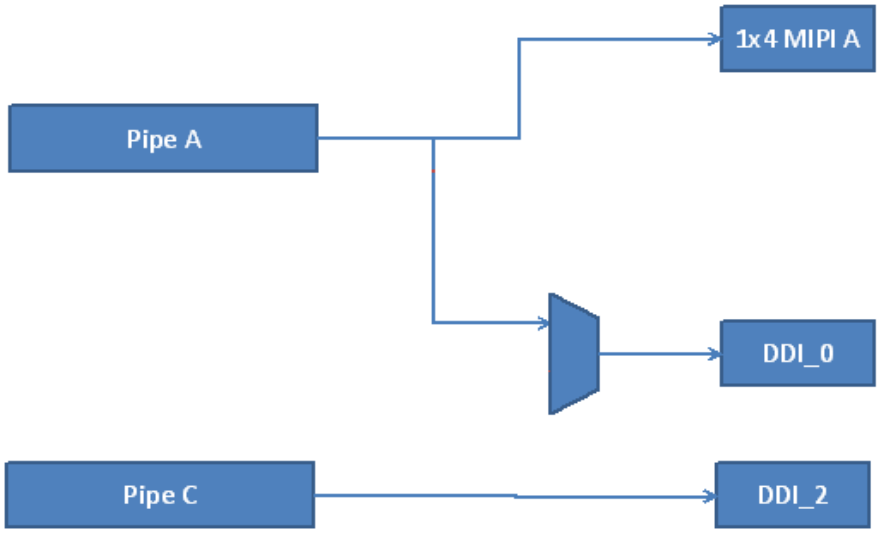


Figure 11. Display Pipe to Port Mapping [VMS-T3]



10.4.1 Digital Display Interfaces

Table 63. SoC Display Configuration

Feature	MIPI DSI	eDP	DP	HDMI/DVI
Number of Ports	2 (x4)	2 (2x4 @2.7GHz)	2 (2x4 @2.7GHz)	2 (2x4 @2.97GHz)
Max Resolution	2560x1600 24bpp @60Hz	3840x2160 24bpp @60Hz	3840x2160 24bpp @60Hz	1920x1080 24bpp @120Hz/ 4K x 2K 24bpp @ 30Hz
Data Rate	8 Gbps	21.6 Gbps	21.6 Gbps	-
Standard	DSI1.01/ DPHY1.00	eDP1.3/eDP1.4	DP1.2	HDMI1.4B
Power gated during S0ix w/display off	Yes	Yes	Yes	Yes

Table 63. SoC Display Configuration

Feature	MIPI DSI	eDP	DP	HDMI/DVI
DRRS (Refresh reduction)	Yes (M/N pair)	Yes (Panel command)	N/A	N/A
Self-Refresh with Frame buffer in Panel	Yes (Command Mode)	Yes (PSR)	No	No
Content-Based backlight control	DPST6.0	DPST6/CABC	N/A	N/A
HDCP wired display	N/A	N/A(ASSR support)	1.4	1.4
PAVP	AES-encrypted buffer, plane control, panic attack			
SEC	All display registers can be accessed by CEC			
HD-Audio	N/A	N/A	Yes	Yes
LPE Audio	N/A	N/A	Yes	Yes
Compressed Audio	N/A	N/A	Yes	Yes

10.4.1.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 65. Display Physical Interfaces Signal Names (2 of 2)

Signal names	Direction Type	Description
MDSI_A_CLKP	O	MIPI Clock output for pipe A
MDSI_A_CLKN	O	MIPI Clock complement output for pipe A
MDSI_A_DP[3:0]	I/O	MIPI Data Lane 3:0 for Pipe A
MDSI_A_DN[3:0]	I/O	MIPI Data Lane 3:0 complement for Pipe A
MDSI_C_CLKP	O	MIPI Clock output for pipe C
MDSI_C_CLKN	O	MIPI Clock complement output for pipe C
MDSI_C_DP[3:0]	I/O	MIPI Data Lane 3:0 for Pipe C
MDSI_C_DN[3:0]	I/O	MIPI Data Lane 3:0 complement for Pipe C
MDSI_A_TE	I/O	Tearing Effect Signal from x4 Pipe A display
MDSI_C_TE	I	Tearing Effect Signal from x4 Pipe C display

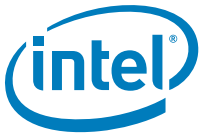


Table 65. Display Physical Interfaces Signal Names (2 of 2)

Signal names	Direction Type	Description
MDSI_DDC_DATA	I/O	DDC Data
MDSI_DDC_CLK	I/O	DDC Clock
MDSI_RCOMP	I/O	MDSI_RCOMP: This is for pre-driver slew rate compensation for the MIPI DSI Interface. An external precision resistor of 150 Ω ±1% should be connected from this pin to ground.

10.4.1.2 Features

10.4.1.2.1 MIPI-DSI

Dual Link interface supports display resolution up to 2560 x 1600p @ 60 Hz with 24b per pixel.

Interface supports maximum of 1Gbps per lane.

Full Frame Buffer Panel

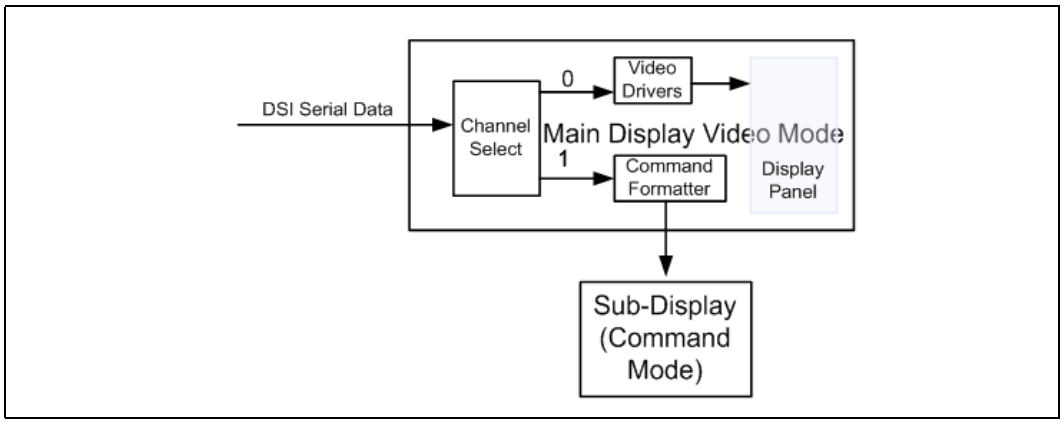
The display controller supports full frame buffer display (also called command-mode display) with optimizations for both SoC power consumption and system power consumption. Full frame buffer panel does not need to be refreshed regularly by a frame buffer in system memory so the path between panel and system memory can be power-managed as much as possible until a new request occurs to update one or more planes that are active in the display pipe.

Sub-Display Support

The display controller supports a sub-display panel that uses a different virtual channel and shares the same interface with the main panel. The pixel data for this sub-display can come from a direct system memory read or it can come from the output at the pipe as described. Sub-display allows, for example, the pixel stream to be updated more frequently or presented in a format and/or resolution that would require software to convert or scale the panel resolution.

One example usage of sub-display is as a view finder for camera. The camera interface unit may output images in a format and resolution that are not read by the sub-display itself or must be blended with camera application graphics.

Figure 12. Sub-Display Connection



Partial Display Mode Support

The display controller supports a partial display mode that utilizes the MIPI command set to transition the panel from normal mode to partial display mode, so a small part of the display panel can be kept active for pixel data. The same panel can switch from full screen mode to a sub-display mode with a handful of scan lines to show time, date, signal strength indicator, etc., to save power for the host processor and display panel.

There are two scenarios:

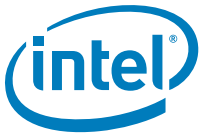
- Type 1 display panel—both full display and partial display operates in command mode.
- Type 2 display panel—full display (normal mode) operate in video mode; partial display operates in command mode. This requires the host processor and display panel to be in sync in transition from normal mode to partial mode after 2 frames from the enter_partial_mode command.

The software driver must implement most of the protocols of transition and send the correct commands to the display panel to start the transition. The software driver must program the display controller to select the buffer for partial display (display pipe output or system memory) and follow the protocol to be in sync with the display panel.

When the display transitions from partial mode to normal mode, it is recommended to turn the display off to avoid tearing effect as in a flow chart in DCS specification.

MIPI DSI Dual-link Mode

The SoC supports MIPI DSI dual-link mode, so that a single display can transmit a single stream of video data across two independent MIPI DSI interfaces. The packetization and timing of each link follows MIPI DSI 1.00 and DPHY 1.00 precisely, but the receiving device, which is a panel or a bridge, can combine the streaming data from two interfaces and display it in a single panel.



There are two types of dual-link panels that the SoC can support: front-back type and pixel alternative type. In the front-back type of panel, the first half of columns of pixels is always transmitted by port A and the second half of columns of pixels is always transmitted by port B.

In the pixel alternative type of panel, odd columns of pixels are always transmitted by port A and even columns of pixels are always transmitted by port B. So the 1st, 3rd, 5th, 7th, etc., pixels are separated at the source and sent in the first interface; the 2nd, 4th, 6th, 8th, etc., pixels are sent in the second interface.When the platform requires a dual-link interface for a large MIPI DSI panel or bridge (usually with resolution larger than 1920x1080 in which a 4-lane interface does not have enough bandwidth), the driver treats dual-link a special port configuration, with special handling of DSI controller but the operation of dual-link mode is consistent with single-link mode for planes and pipe operations. The system interface with upper level of SW doesn't need to change, like flip mechanism, interrupt, and so on.

LVDS Panel Support

An external MIPI DSI-to-LVDS bridge device is required to connect the display controller to a LVDS panel. A bridge device is used for larger panels.

10.4.1.3 High Definition Multimedia Interface

The High-Definition Multimedia Interface (HDMI) is provided for transmitting digital audio and video signals from DVD players, set-top boxes and other audiovisual 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. HDMI display interface connecting the SoC and display devices utilizes transition minimized differential signaling (TMDS) to carry audiovisual information through the same HDMI cable.

HDMI includes three separate communications channels: TMDS, DDC, and the optional CEC (consumer electronics control) (not supported by the SoC). As shown in Figure 13 the HDMI cable carries four differential pairs that make up the TMDS data and clock channels. These channels are used to carry video, audio, and auxiliary data. In addition, HDMI carries a VESA DDC. The DDC is used by an HDMI Source to determine the capabilities and characteristics of the sink.

Audio, video and auxiliary (control/status) data is transmitted across the three TMDS data channels. The video pixel clock is transmitted on the TMDS clock channel and is used by the receiver for data recovery on the three data channels. The digital display data signals driven natively through the SoC are AC coupled and needs level shifting to convert the AC coupled signals to the HDMI compliant digital signals.

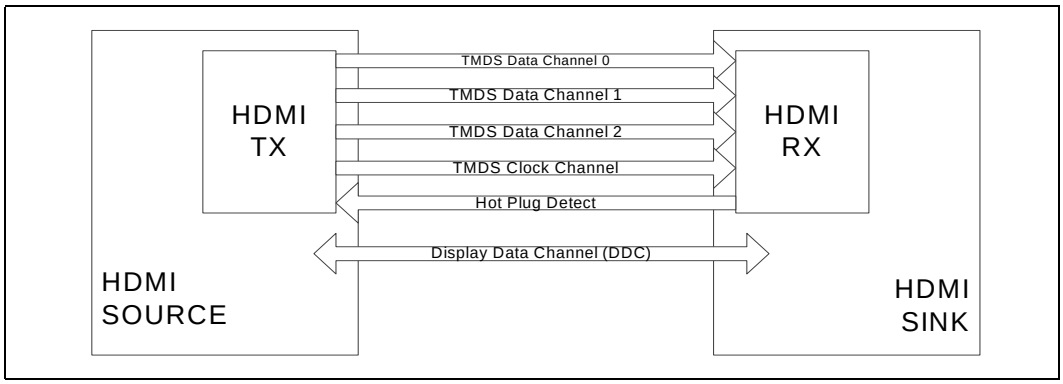
The SoC HDMI interface is designed as per the High-Definition Multimedia Interface Specification 1.4. The SoC supports High-Definition Multimedia Interface Compliance Test Specification 1.4.

10.4.1.3.1 Stereoscopic Support on HDMI

SoC display supports HDMI 1.4 3D video formats. If the HDMI panel is detected to support 3D video format then the SW driver will program Pipe2dB for the correct pipe timing parameters.

The left and right frames can be loaded from independent frame buffers in the main memory. Depending on the input S3D format, the display controller can be enabled do perform frame repositioning, image scaling, line interleaving.

Figure 13. HDMI Overview



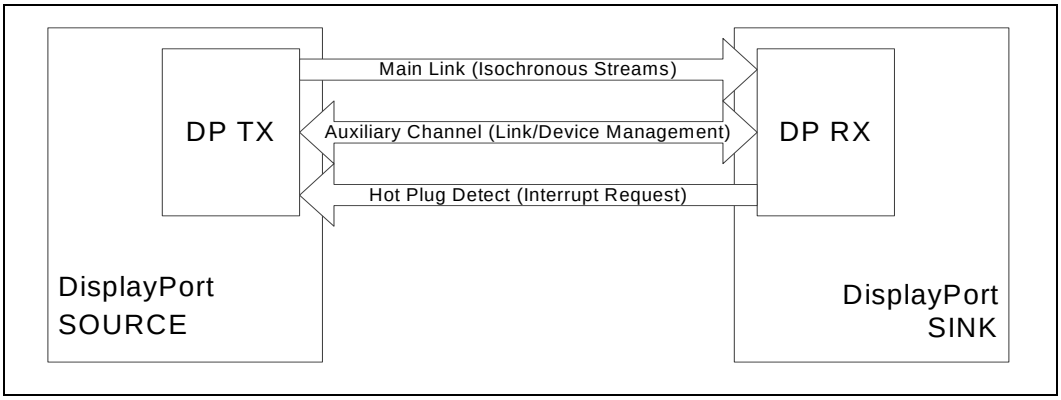
10.4.1.4 Display Port

Display Port is a digital communication interface that utilizes differential signalling to achieve a high bandwidth bus interface designed to support connections between PCs and monitors, projectors, and TV displays. Display Port is also suitable for display connections between consumer electronics devices such as high definition optical disc players, set top boxes, and TV displays.

A Display Port consists of a Main Link, Auxiliary channel, and a Hot Plug Detect signal. The Main Link is a uni-directional, high-bandwidth, and low latency channel used for transport of isochronous data streams such as uncompressed video and audio. The Auxiliary Channel (AUX CH) is a half-duplex bidirectional channel used for link management and device control. The Hot Plug Detect (HPD) signal serves as an interrupt request for the sink device.

The SoC supports DisplayPort Standard Version 1.2.

Figure 14. DisplayPort* Overview



10.4.1.5 Embedded DisplayPort (eDP)

Embedded DisplayPort (eDP) is a embedded version of the DisplayPort standard oriented towards applications such as notebook and All-In-One PCs. eDP is supported only on Digital Display Interfaces 0 and/or 1. Like DisplayPort, Embedded DisplayPort also consists of a Main Link, Auxiliary channel, and a optional Hot Plug Detect signal.

Each eDP port can be configured for up-to 4 lanes.

The SoC supports Embedded DisplayPort Standard Version 1.3 and v1.4.

10.4.1.5.1 DisplayPort Auxiliary Channel

A bidirectional AC coupled AUX channel interface replaces the I^2C for EDID read, link management and device control. I^2C-to-Aux bridges are required to connect legacy display devices.

10.4.1.5.2 Hot-Plug Detect (HPD)

The SoC supports HPD for Hot-Plug sink events on the HDMI and DisplayPort interfaces.

10.4.1.5.3 Integrated Audio over HDMI and DisplayPort

SoC can support two audio streams on DP/HDMI ports. Each stream can be programmable to either DDI port. HDMI/DP audio streams can be sent with video streams as follows.

LPE mode: In this mode the uncompressed or compressed audio sample buffers are generated either by OS the audio stack or by audio Lower Power Engine (LPE) and stored in system memory.The display controller fetches audio samples from these buffers, forms an SPDIF frame with VUCP and preamble (if needed), then sends out with video packets.

10.4.1.5.4 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, etc.) and the sink (panels, monitor, and TV). The SoC supports HDCP 1.4/2.1 for content protection over wired displays (HDMI, DisplayPort and Embedded DisplayPort).

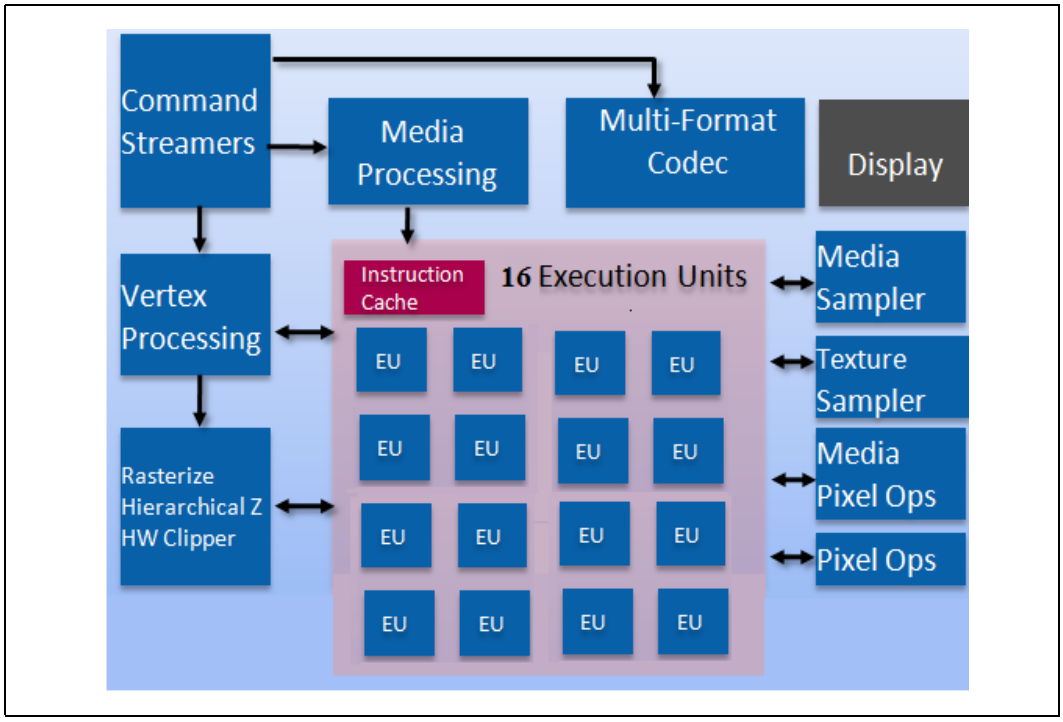
10.5 References

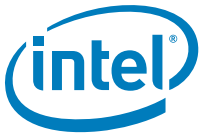
- High-Definition Multimedia Interface Specification, Version 1.4b
- High-bandwidth Digital Content Protection System, Revision 1.4
- VESA DisplayPort Standard, Version 1.2
- VESA Embedded DisplayPort Standard, Version 1.4

10.6 3D Graphics and Video

The SoC implements a derivative of the Generation 8 LP graphics engine which consists of rendering engine and bit stream encoder/decoder engine. The rendering engine is used for 3D rendering, media compositing and video encoding. The Graphics engine is built around sixteen execution units (EUs).

Figure 15. 3D Graphics Block Diagram





10.7 Features

The 3D graphics pipeline architecture simultaneously operates on different primitives or on different portions of the same primitive. All the cores are fully programmable, increasing the versatility of the 3D Engine. The Gen 8.0 LP 3D engine provides the following performance and power-management enhancements:

- Hierarchal-Z
- Video quality enhancements

10.7.1 3D Engine Execution Units

- The EUs perform 128-bit wide execution per clock.
- Support SIMD8 instructions for vertex processing and SIMD16 instructions for pixel processing.

10.7.2 3D Pipeline

10.7.2.1 Vertex Fetch (VF) Stage

The VF stage executes 3DPRIMITIVE commands. Some enhancements have been included to better support legacy D3D APIs as well as SGI OpenGL*.

10.7.2.2 Vertex Shader (VS) Stage

The VS stage performs shading of vertices output by the VF function. The VS unit produces an output vertex reference for every input vertex reference received from the VF unit, in the order received.

10.7.2.3 Geometry Shader (GS) Stage

The GS stage receives inputs from the VS stage. Compiled application-provided GS programs, specifying an algorithm to convert the vertices of an input object into some output primitives. For example, a GS shader may convert lines of a line strip into polygons representing a corresponding segment of a blade of grass centered on the line. Or it could use adjacency information to detect silhouette edges of triangles and output polygons extruding out from the edges.

10.7.2.4 Clip Stage

The Clip stage performs general processing on incoming 3D objects. However, it also includes specialized logic to perform a Clip Test function on incoming objects. The Clip Test optimizes generalized 3D Clipping. The Clip unit examines the position of incoming vertices, and accepts/rejects 3D objects based on its Clip algorithm.

10.7.2.5 Strips and Fans (SF) Stage

The SF stage performs setup operations required to rasterize 3D objects. The outputs from the SF stage to the Windower stage contain implementation-specific information required for the rasterization of objects and also supports clipping of primitives to some extent.

10.7.2.6 Windower/IZ (WIZ) Stage

The WIZ unit performs an early depth test, which removes failing pixels and eliminates unnecessary processing overhead.

The Windower uses the parameters provided by the SF unit in the object-specific rasterization algorithms. The WIZ unit rasterizes objects into the corresponding set of pixels. The Windower is also capable of performing dithering, whereby the illusion of a higher resolution when using low-bpp channels in color buffers is possible. Color dithering diffuses the sharp color bands seen on smooth-shaded objects.

10.7.3 Video Engine

The video engine is part of the Intel Processor Graphics for image processing, play-back and transcode of Video applications. Processor Graphics video engine has a dedicated fixed hardware pipe-line for high quality decode and encode of media content. This engine supports Full HW acceleration for decode of AVC/H.264, VC-1 and MPEG -2 contents along with encode of MPEG-2 and AVC/H.264 apart from various video processing features. The new Processor Graphics Video engine adds support for processing features such as frame rate conversion, image stabilization and gamut conversion.

10.8 VED (Video Encode/Decode)

The video engine is part of the Intel Processor Graphics for image processing, play-back and transcode of Video applications. Processor Graphics video engine has a dedicated fixed hardware pipe-line for high quality decode and encode of media content.

10.8.1 Features

The features for the Video decode hardware accelerator in SoC are:

- VED core can be configured on a time division multiplex basis to handle single, dual and multi-stream HD decoding/encoding.
- VED provides full hardware acceleration Decode/Encode support below Media formats

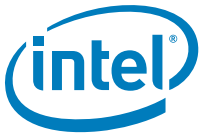


Table 66. Hardware Accelerated Video Decode/Encode Codec Support

Codec Format	Decode Level	Encode Level
HEVC (H.265)	1080p60, 4Kx2Kp30	-
H.264	HP L5.1 up to 1080p120, 4Kx2Kp30	HP L5.1 up to 1080p120, 4Kx2Kp30
MPEG2	MP HL (1080p30)	1080p30
MVC	1080p30	1080p30
VC-1	AP L4 (1080p30)	
WMV9	MP HL (1080p30)	
Xvid, DivX		
SVC	Up to 1080p30	Up to 1080p30
VP8	1080p60	1080p60
MPEG4P2	1080p30	
H.263		
Sorenson		
Real Video		
JPEG/MJPEG		
OGG Theora		

10.9 Register Map

For more information on Graphics, Video and Display registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§

11 PCI Express 2.0

There are up to two PCI Express root ports, each supporting the *PCI Express* Base Specification*, Rev. 2.0 at a maximum 5 GT/s signaling rate. The root ports can be configured to support a diverse set of lane assignments.

11.1 Signal Descriptions

See Chapter 2, "Physical Interfaces" for additional details.

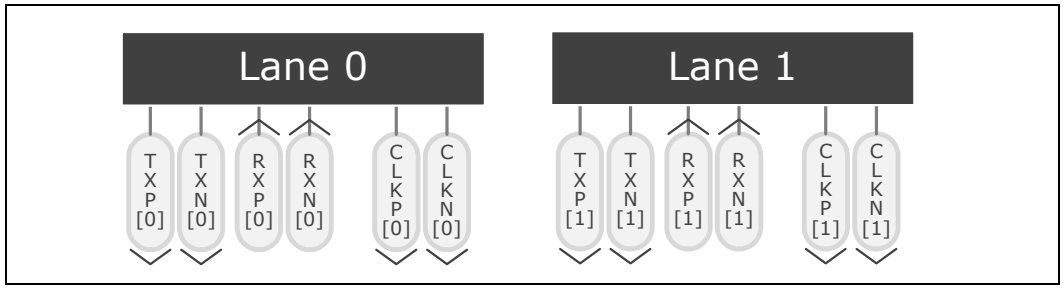
The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 67. Signals

Signal Name	Direction /Type	Description
PCIE_TXP[1:0] **PCIE_TXN[1:0]**	O PCIe	**PCI Express* Transmit** PCI Express* Ports 1:0 transmit pair (P and N) signals. Each pair makes up the transmit half of a lane.
PCIE_RXP[1:0] **PCIE_RXN[1:0]**	I PCIe	**PCI Express* Receive:** PCI Express* Ports 1:0 receive pair (P and N) signals. Each pair makes up the receive half of lane.
PCIE_CLKREQ[1:0]_N	IO	**PCI Express* Clock Request** Used for devices that need to request one of the output clocks. Each clock request maps to the matching clock output (e.g., PCI_CLKREQ[0] maps to PCIE_CLKP/N[0]). **NOTE:** These signals are muxed and may be used by other functions.
P_RCOMP_P **P_RCOMP_N**	I/O	These pins are used to connect the external resistors used for Rcomp. Refer to PDG for details.

Figure 16. PCIe* 2.0 Lane 0 Signal Example



11.2 Features

- Conforms to *PCI Express* Base Specification*, Rev. 2.0
- 5.0 or 2.5 GT/s operation per root port
- Virtual Channel support for VC0 only
- x1, x2 link widths (auto negotiated)
- Flexible Root Port configuration options
  - (1) x2's
  - (1) x2 plus (1) x1's
  - (2) x1
- Interrupts and Events
  - Legacy (INT*x*) and MSI Interrupts
  - General Purpose Events
  - Express Card Hot Plug Events
  - System Error Events
- Power Management
  - Link State support for L0s, L1 and L2
  - Powered down in ACPI S3 state - L3

11.2.1 Root Port Configurations

Depending on SKU, there are up to two possible lane assignments for root ports 1-2.

Root port configurations are set by SoftStraps stored in SPI flash, and the default option is "(2) x1". Links for each root port will train automatically to the maximum possible for each port.

Note: x2 link widths are not common. Most devices will only train to x1.

Note: PCI functions in PCI configuration space are disabled for root ports not available.

11.2.2 Interrupts and Events

A root port is capable of handling interrupts and events from an end point device. A root port can also generate its own interrupts for some events, including power management and hot plug events, but also including error events.

There are two interrupt types a root port will receive from an end point device: INTx (legacy), and MSI. MSI's are automatically passed upstream by the root port, just as other memory writes would be. INTx messages are delivered to the Legacy block's interrupt router/controller by the root port.

Events and interrupts that are handled by the root port are shown with the possible interrupts they can deliver to the interrupt decoder/router.

Table 68. Possible Interrupts Generated From Events/Packets

Packet/Event	Type	INTx	MSI	SERR	SCI	SMI	GPE
INTx	Packet	X	X				
PM_PME	Packet	X	X				
Power Management (PM)	Event	X	X		X	X	
Hot Plug (HP)	Event	X	X		X	X	
ERR_CORR	Packet			X			
ERR_NONFATAL	Packet			X			
ERR_FATAL	Packet			X			
Internal Error	Event			X			
VDM	Packet						X

NOTE: Table 68 lists the possible interrupts and events generated based on Packets received, or events generated in the root port. Configuration needed by software to enable the different interrupts as applicable.

When INTx interrupts are received by an end point, they are mapped to the following interrupts and sent to the interrupt decoder/router in the iLB:

Table 69. Interrupt Generated for INT[A-D] Interrupts

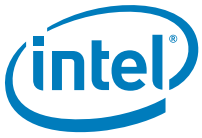
	INTA	INTB	INTC	INTD
Root Port 1	INTA#	INTB#	INTC#	INTD#
Root Port 2	INTD#	INTA#	INTB#	INTC#

Note: Interrupts generated from events within the root port are not swizzled.

11.2.2.1 Express Card Hot Plug Events

Express Card Hot Plug is available based on Presence Detection for each root port.

Note: A full Hot Plug Controller is not implemented.



Presence detection occurs when a PCI Express* device is plugged in and power is supplied. The physical layer will detect the presence of the device, and the root port will set the SLSTS.PDS and SLSTS.PDC bits.

When a device is removed and detected by the physical layer, the root port will clear the SLSTS.PDS bit, and set the SLSTS.PDC bit.

Interrupts can be generated by the root port when a hot plug event occurs. A hot plug event is defined as the transition of the SLSTS.PDC bit from 0 to 1. Software can set the SLCTL.PDE and SLTCTL.HPE bits to allow hot plug events to generate an interrupt.

If SLCTL.PDE and SLTCTL.HPE are both set, and STSTS.PDC transitions from 0 to 1, an interrupt will be generated.

11.2.2.2 System Error (SERR)

System Error events are support by both internal and external sources. See the PCI Express* Base Specification, Rev. 2.0 for details.

11.2.3 Power Management

Each root port's link supports L0s, L1, and L2/3 link states per PCI Express* Base Specification, Rev. 2.0. L2/3 is entered on entry to S3.

11.3 References

PCI Express Base Specification*, Rev. 2.0

11.4 Register Map

For more information on PCI Express 2.0 registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§

12 MIPI-Camera Serial Interface (CSI) and ISP

The MIPI CSI and controller front end interfaces with three sensors and is capable of simultaneously acquiring three streams, one from each sensor. These three streams are presented to the ISP.

12.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 70. CSI Signals

Signal Name	Direction	Description
MCSI1_CLKP/N	I	**Clock Lane:** MIPI CSI input clock lane 0 for port 1.
MCSI1_DP/N[3:0]	I	**Data Lanes:** Four MIPI CSI Data Lanes (0-3) for port 1. Lanes 2 and 3 can optionally used as data lanes for port 3.
MCSI2_CLKP/N	I	**Clock Lane:** MIPI CSI input clock lane 0 for port 2.
MCSI2_DP/N[1:0]	I	**Data Lane:** Two MIPI CSI Data Lanes for port 2.
MCSI3_CLKP/N	I	**Clock Lane:** MIPI CSI input clock lane 0 for port 3.
MCSI_RCOMP	I/O	**Resistor Compensation:** This is for pre-driver slew rate compensation for the MIPI CSI Interface. See design guide for details.

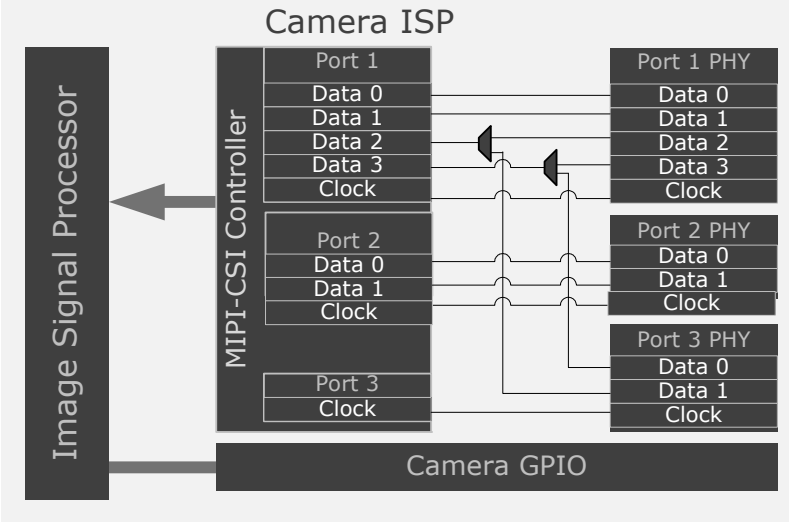
Table 71. GPIO Signals (Sheet 1 of 2)

Signal Name	Direction /Type	Description
MCSI_GPIO[00]	I/O	Output from shutter switch when its pressed halfway. This switch state is used to trigger the Auto focus LED for Xenon Flash or Torch mode for LED Flash
MCSI_GPIO[01]	I/O	Output from shutter switch when its pressed full way. This switch state is used to trigger Xenon flash or LED Flash
MCSI_GPIO[02]	I/O	Active high control signal to Xenon Flash to start charging the Capacitor
MCSI_GPIO[03]	I/O	Active low output from Xenon Flash to indicate that the capacitor is fully charged and is ready to be triggered

Table 71. GPIO Signals (Sheet 2 of 2)

Signal Name	Direction /Type	Description
MCSI_GPIO[04]	I/O	Active high Xenon Flash trigger / Enables Torch Mode on LED Flash IC
MCSI_GPIO[05]	I/O	Enables Red Eye Reduction LED for Xenon / Triggers STROBE on LED Flash IC /
MCSI_GPIO[06]	I/O	Camera Sensor 0 Strobe Output to SoC to indicate beginning of capture / Active high signal to still camera to power down the device.
MCSI_GPIO[07]	I/O	Camera Sensor 1 Strobe Output to SoC to indicate beginning of capture / Active high signal to still camera to power down the device.
MCSI_GPIO[08]	I/O	Active high signal to video camera to power down the device.
MCSI_GPIO[09]	I/O	Active low output signal to reset digital still camera #0.
MCSI_GPIO[10]	I/O	Active low output signal to reset digital still camera #1
MCSI_GPIO[11]	I/O	Active low output signal to reset digital video camera

Figure 17. Camera Connectivity



Note: Not all MCSI_GPIO signals are supported on VMS T3, please refer to VMS T3 VGPIO table for the supported signals.

12.2 Features

- Integrated MIPI-CSI 2.0 interface
- Image Signal Processor (ISP) with DMA and local SRAM

- Imaging data is received by the MIPI-CSI interface and is relayed to the ISP for processing
- Up to six MIPI-CSI 2.0 data lanes
  - Each lane can operate at up to 1.5Gbp/s. resulting in roughly 1.2 Gbp/s of actual pixels
- The MIPI-CSI interface supports lossless compressed image streams to increases the effective bandwidth without losing data
- Up to 13MP sensors supported, and full HD 1080p60
  - Can also support Stereo HD 1080p30

12.2.1 Imaging Capabilities

The following table summarizes imaging capabilities.

Table 72. Imaging Capabilities

Feature	Capabilities
Sensor interface	Configurable MIPI-CSI2 interfaces. 3 sensors: x2, x2, x2 or x1 x2, x3 2 sensors: x4, x2
Simultaneous sensors	Up to 3 simultaneous sensors
2D Image capture	13MP ZSL @ 18fps
2D video capture	Up to 1080p60
Input formats	RAW 8, 10, 12, 14, RGB444, 565, 888, YUV420, 422, JPEG.
Output formats)	YUV422, YUV420, RAW
Special Features	Image and video stabilization Low light noise reduction Burst mode capture Memory to memory processing 3A (Auto Exposure (AE), Auto White Balance (AWB) and Auto Focus (AF)) High Dynamic Range (HDR) Multi-focus Zero shutter lag

12.2.2 Simultaneous Acquisition

SoC will support on-the-fly processing for only one image at a time. While this image is being processed on-the-fly, images from the other two cameras are saved to DRAM for later processing.

12.2.3 Primary Camera Still Image Resolution

Maximum still image resolution for the primary camera in post-processing mode is limited by the resolution of the sensors. Currently 13Mpixel sensors are supported.

Higher resolution, or higher frame rates are supported as long as the product of resolution and frame rate does not exceed 235 Mpixels/s (= 13 Mpixels * 18 fps).

Maximum primary camera on-the-fly stereoscopic still image resolution for primary camera is 8 Mpixel for each of the left and right images at 18 fps. The number of Mpixels can be increased by decreasing the frame rate.

12.2.4 Burst Mode Support

The SoC supports capturing multiple images back to back at maximum sensor resolution. At least 5 images must be captured in burst mode. The maximum number of images that can be so captured is limited only by available system memory. These images need not be processed on-the-fly.

12.2.5 Continuous Mode Capture

SoC supports capturing images and saving them to DRAM in a ring of frame buffers continuously at maximum sensor resolution. This adds a round trip to memory for every frame and increases the bandwidth requirements.

12.2.6 Secondary Camera Still Image Resolution

Maximum secondary camera still image resolution is 4 Mpixel at 15 fps.

12.2.7 Primary Camera Video Resolution

Maximum primary camera video resolution is 1080p60.

Maximum primary camera dual video resolution is 1080p30.

12.2.8 Secondary Camera Video Resolution

Maximum secondary camera video resolution is 1080p30.

12.2.9 Bit Depth

Capable of processing 14-bit images at the stated performance levels.

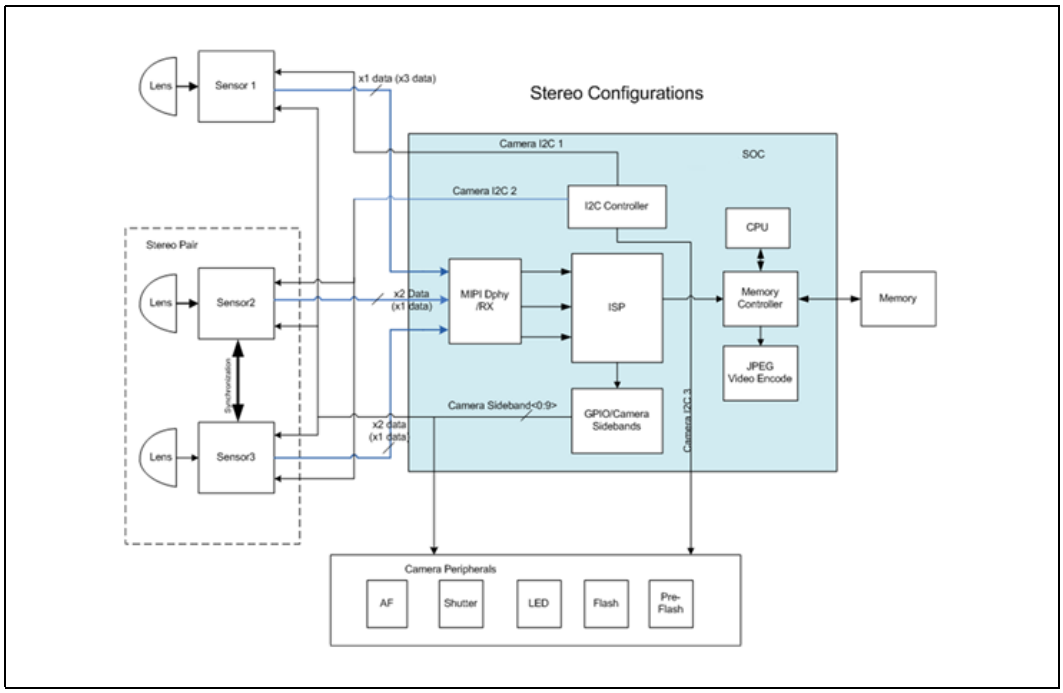
Capable of processing 18-bit images at half the performance levels, i.e. process on-the-fly 13 Mpixel 18-bit images at 7 fps instead of 15 fps.

Capable of processing up to 18-bit precision.

The higher precision processing will be employed mainly for high dynamic range imaging (HDR).

12.3 Imaging Subsystem Integration

Figure 18. Image Processing Components



12.3.1 CPU Core

The CPU core augments the signal processing capabilities of the hardware to perform post-processing on images such as auto focus, auto white balance, and auto exposure. The CPU also runs the drivers that control the GPIOs and I^2C for sensor control.

12.3.2 Imaging Signal Processor (ISP)

The ISP (Imaging Signal Processor) includes a 64-way vector processor enabling high quality camera functionality. Key features include support of three camera sensors.

12.3.2.1 MIPI-CSI-2 Ports

The SoC has three MIPI clock lanes and six MIPI data lanes. The Analog Front End (AFE) and Digital Physical Layer (DPHY) take these lanes and connects them to three virtual ports. Two data lanes are dedicated to each of the rear facing cameras and the remaining data lane is connected to the front facing camera. The MIPI interfaces follow the MIPI-CSI-2 specifications as defined by the MIPI Alliance. They support YUV420,

YUV422, RGB444, RGB555, RGB565, and RAW 8b/10b/12b. Both MIPI ports support compression settings specified in MIPI-CSI-2 draft specification 1.01.00 Annex E. The compression is implemented in Hardware with support for Predictor 1 and Predictor 2. Supported compression schemes:

- 12-8-12
- 12-7-12
- 12-6-12
- 10-8-10
- 10-7-10
- 10-6-10

The data compression schemes above use an X-Y-Z naming convention where X is the number of bits per pixel in the original image, Y is the encoded (compressed) bits per pixel and Z is the decoded (uncompressed) bits per pixel.

12.3.2.2 I^2C for Camera Interface

The platform supports three (3) I^2C ports for the camera interface. These ports are used to control the camera sensors and the camera peripherals such as flash LED and lens motor.

12.3.2.3 Camera Sideband for Camera Interface

Twelve (12) GPIO signals are allocated for camera functions, refer to Table 71 for signal names. These GPIOs are multiplexed and are available for other usages without powering on the ISP. The ISP provides a timing control block through which the GPIOs can be controlled to support assertion, de-assertion, pulse widths and delay. The configuration below of camera GPIOs is just an example of how the GPIOs can be used. Several of these functions could be implemented using I^2C, depending on the sensor implementation for the platform.

- Sensor Reset signals

  —Force hardware reset on one or more of the sensors.
- Sensor Single Shot Trigger signal

  —Indicate that the target sensor needs to send a full frame in a single shot mode, or to capture the full frame for flash synchronization.
- PreLight Trigger signal

  —Light up a pilot lamp prior to firing the flash for preventing red-eye.
- Flash Trigger signal

  —Indicate that a full frame is about to be captured. The Flash fires when it detects an assertion of the signal.
- Sensor Strobe Trigger signal

  —Asserted by the target sensor to indicate the start of a full frame, when it is configured in the single shot mode, or to indicate a flash exposed frame for flash synchronization.

12.4 Functional Description

At a high level, the Camera Subsystem supports the following modes:

- Preview
- Image capture
- Video capture

12.4.1 Preview Mode

Once the ISP and the camera subsystem is enabled, the ISP goes into the preview mode where very low resolution frames, such as VGA/480p (programmable), are being processed.

12.4.2 Image Capture

During the image capture mode, the camera subsystem can acquire at a peak throughput of 13 Mpixels @ 18fps. While doing this, it continues to output preview frames simultaneously.

- The ISP can output RAW, RGB or YUV formats. The ISP can capture one full frame at a time or perform burst mode capture, where up to five full back-to-back frames are recorded.
- The ISP will not limit the number of back-to-back full frames captured, but the number is programmable and determined on how much memory can be allocated dynamically.
- The ISP can process all the frames on the fly and writes to memory only after fully processing the frames, without requiring download of any part of the frame for further processing.
  - —The exceptions to this approach are image stabilization and some other advanced functions requiring temporal information over multiple frames.

The ISP can support image stabilization in image capture model.

- The ISP initially outputs preview frames.
- When the user decides to capture the picture, image stabilization is enabled. The ISP checks the previous frame for motion and compensates for it appropriately.

Auto Exposure (AE), Auto Focus (AF), and Auto White Balance (AWB), together known as 3A, are implemented in the CPU to provide flexibility.

12.4.3 Video Capture

During video recording, the ISP can capture video up to 1080p @ 60 fps and output preview frames concurrently. The ISP output video frames to memory in YUV420 or YUV422 format.

12.4.4 ISP Overview

The Camera Subsystem consists of 2 parts, the hardware subsystem and a software stack that implements the ISP functionality on top of this hardware.

The core of the ISP is a vector processor. The vector processor is supported by the following components:

- Interfaces for data and control
- A small input formatter that parallelizes the data
- A scalar (RISC) processor, for system control and low-rate processing
- An accelerator for scaling, digital zoom, and lens distortion correction
- A DMA engine transfers large amounts of data such as input and output image data or large parameter sets between LPDDR2 and the ISP block.

12.4.5 Memory Management Unit (MMU)

The camera subsystem has capabilities to deal with a virtual address space, since a contiguous memory range in the order of 16–32MB cannot be guaranteed by the OS.

12.4.5.1 Interface

The MMU performs the lookup required for address translation from a 32-bit virtual address to 36-bit physical address. The lookup tables are stored external to the system. The MMU performs the lookup through a master interface without burst support that is connected to the Open Core Protocol (OCP) master of the subsystem. The MMU configuration registers can be accessed through a 32-bit Core I/O (CIO) slave interface. Additionally there is a 32-bit CIO slave interface connected to the address translator.

12.5 MIPI-CSI-2 Receiver

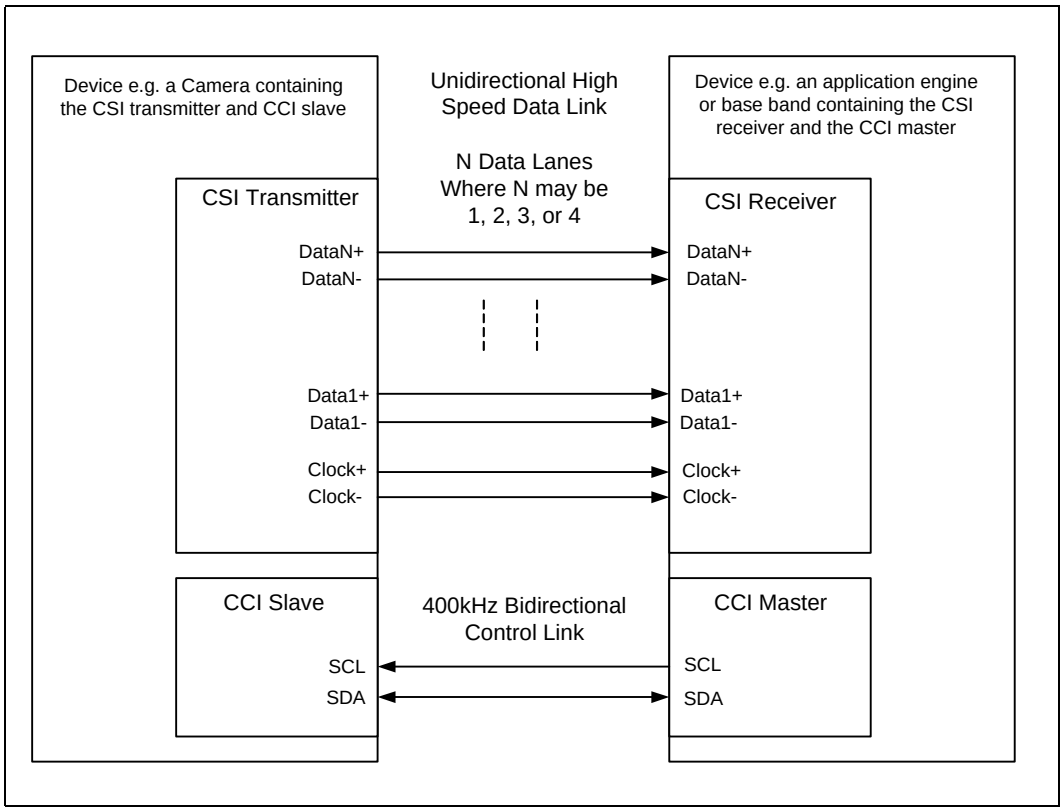
MIPI-CSI-2 devices are camera serial interface devices. They are categorized into two types, a CSI transmitter device with Camera Control Interface (CCI) slave and CSI receiver device with CCI master.

Data transfer by means of MIPI-CSI is unidirectional that is, from transmitter to receiver. CCI data transfer is bidirectional between the CCI slave and master.

Camera Serial Interface Bus (CSI) is a type of serial bus that enables transfer of data between a Transmitter device and a receiver device. The CSI device has a point-to-point connections with another CSI device by means of D-PHYs and as shown in Figure 19.

Similarly, CCI (Camera Control Interface bus) is a type of serial bus that enables transfer back and forth between the master CCI and a Slave CCI Unit.

Figure 19. MIPI-CSI Bus Block Diagram



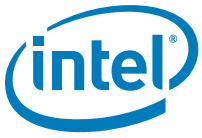
D-PHY data lane signals are transferred point-to-point differentially using two signal lines and a clock lane. There are two signaling modes, a high speed mode that operates up-to 1500Mbs and a low power mode that works at 10Mbs. The mode is set to low power mode and a stop state at start up/power up. Depending on the desired data transfer type, the lanes switch between high and low power modes.

The CCI interface consists of an I^2C bus which has a clock line and a bidirectional data line.

The MIPI-CSI-2 devices operate in a layered fashion. There are 5 layers identified at the receiver and transmitter ends.

MIPI-CSI-2 Functional Layers:

- **PHY Layer**:
  - An embedded electrical layer sends and detects start of packet signalling and end of packet signalling on the data lanes. It contains a serializer and deserializer unit to interface with the PPI / lane management unit. There is also a clock divider unit to source and receive the clock during different modes of operation.



- **PPI/Lane Management Unit**:
  - This layer does the lane buffering and distributes the data in the lanes as programmed in a round robin manner and also merges them for the PLI/Low Level Protocol unit.
- **PLI/Low Level Protocol Unit**:
  - This layer packetizes as well as de-packetizes the data with respect to channels, frames, colors and line formats. There are ECC generator and corrector units to recover the data free from errors in the packet headers. There is also a CRC checker or CRC generator unit to pack the payload data with CRC checksum bits for payload data protection.
- **Pixel/Byte to Byte/Pixel Packing Formats**:
  - Conversion of pixel formats to data bytes in the payload data is done depending on the type of image data supported by the application. It also re-converts the raw data bytes to pixel format understandable to the application layer.
- **Application**:
  - Depending on the type of formats, camera types, capability of the camera used by the transmitter, the application layer recovers the image formats and reproduces the image in the display unit. It also works on de-framing the data into pixel-to-packing formats. High level encoding and decoding of image data is handled in the application unit.

12.5.1 MIPI-CSI-2 Receiver Features

CSI Features:

- Compliant to CSI-2 MIPI specification for Camera Serial Interface (Version 1.00)
- Supports standard D-PHY receiver compliant to the MIPI Specification
- Supports PHY data programmability up to four lanes.
- Supports PHY data time-out programming.
- Has controls to start and re-start the CSI-2 data transmission for synchronization failures and to support recovery.
- The ISP may not support all the data formats that the CSI-2 receiver can handle.

  —Refer to Table 72 for formats supported by the ISP
- Supports all generic short packet data types.
- Single Image Signal Processor interface for pixel transfers to support multiple image streams for all virtual channel numbers.

D-PHY Features:

- Supports synchronous transfer in high speed mode with a bit rate of 80-1500Mb/s
- Supports asynchronous transfer in low power mode with a bit rate of 10Mb/s.
- Differential signalling for HS data
- Spaced one-hot encoding for Low Power [LP] data
- Data lanes support transfer of data in high speed as well as low power modes.

- Supports ultra low power mode, escape mode, and high speed mode
- Hasa clock divider unit to generate clock for parallel data reception and transmission from and to the PPI unit.
- Activates and disconnects high speed terminators for reception and control mode.
- Activates and disconnects low power terminators for reception and transmission.

12.6 Register Map

For more information on MIPI-Camera Serial Interface (CSI) and ISP registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§

13 SoC Storage

13.1 SoC Storage Overview

13.1.1 Storage Control Cluster (eMMC, SDIO, SD)

The SCC consists of SDIO, SD and eMMC controllers to support mass storage and IO devices.

- One eMMC 5.0 interface
- One SD 3.0 interface
- One SDIO 3.0 interface

13.2 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 73. eMMC Signals (Sheet 1 of 2)

Signal Name	Direction /Type	Description
MMC1_CLK	I/O/GPIO	**eMMC Clock** The frequency may vary between 25 and 200MHz.
MMC1_D[7:0]	I/O/GPIO	**eMMC Port Data bits 0 to 7** Bidirectional port used to transfer data to and from eMMC device. By default, after power up or reset, only D[0] is used for data transfer. A wider data bus can be configured for data transfer, using either D[0]-D[3] or D[0]-D[7], by the MultiMedia Card controller. The MultiMedia Card includes internal pull-ups for data lines D[1]-D[7]. Immediately after entering the 4-bit mode, the card disconnects the internal pull ups of lines D[1], D[2], and D[3]. Correspondingly, immediately after entering to the 8-bit mode the card disconnects the internal pull-ups of lines D[1]-D[7].
MMC1_CMD	I/O/GPIO	**eMMC Port Command** This signal is used for card initialization and transfer of commands. It has two modes—open-drain for initialization, and push-pull for fast command transfer.

Table 73. eMMC Signals (Sheet 2 of 2)

Signal Name	Direction /Type	Description
MMC1_RCOMP	I/O/GPIO	**eMMC RCOMP** This signal is used for pre-driver slew rate compensation.
MMC1_RST_N	I/O/GPIO	**eMMC Reset Signals** Active low to reset.
MMC1_RCLK	I/GPIO	**eMMC Return Clock Signals**

Table 74. SDIO Signals

Signal Name	Direction /Type	Description
SD2_CLK	I/O/GPIO	**SDIO Clock** The frequency may vary between 25 and 200MHz.
SD2_D[2:0]	I/O/GPIO	**SDIO Port Data bits 0 to 2** Bidirectional port used to transfer data to and from SDIO device. By default, after power up or reset, only D[0] is used for data transfer. A wider data bus can be configured for data transfer, using D[0]-D[3].
SD2_D[3]_CD_N	I/O/GPIO	**SDIO Port Data bit 3** Bidirectional port used to transfer data to and from the SDIO device. Also, **Card Detect**. Active low when device is present.
SD2_CMD	I/O/GPIO	**SDIO Port Command** This signal is used for card initialization and transfer of commands. It has two modes—open-drain for initialization, and push-pull for fast command transfer.

Table 75. SD Signals (Sheet 1 of 2)

Signal Name	Direction /Type	Description
SD3_CLK	I/O/GPIO	**SD Card Clock** The frequency may vary between 25 and 200 MHz.
SD3_D[3:0]	I/O/GPIO	**SD Card Data bits 0 to 3** Bidirectional port used to transfer data to and from SD/MMC card. By default, after power up or reset, only D[0] is used for data transfer. A wider data bus can be configured for data transfer, using D[0]-D[3].
SD3_CD_N	I/O/GPIO	**SD Card Detect** Active low when a card is present. Floating (pulled high with internal PU) when a card is not present.
SD3_CMD	I/O/GPIO	**SD Card Command** This signal is used for card initialization and transfer of commands. It has two modes—open-drain for initialization, and push-pull for fast command transfer.

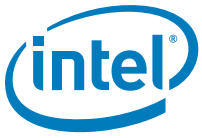


Table 75. SD Signals (Sheet 2 of 2)

Signal Name	Direction /Type	Description
SD3_1P8EN	I/O/GPIO	**SD Card 1.8V Enable** Controls the voltage of the SD Card, the default is low (3.3V). The voltage is 1.8V when this signal is high.
SD3_RCOMP	I/O/GPIO	**SD Card RCOMP** This signal is used for pre-driver slew rate compensation.
SD3_PWREN_N	I/O/GPIO	**SD Card Power Enable** This signal is used to enable power on a SD device.
SD3_WP	I/O/GPIO	**SD Card Write Protect** Active high to protect from write.

13.3 Features

13.3.1 SDIO/SD Interface Features

- Host clock rate variable between 0 and 200 MHz
- Up to 800 Mbits per second data rate using 4 parallel data lines (SDR104 mode).
- Transfers the data in 1 bit and 4 bit SD modes.
- Transfers the data in following UHS-I modes (SDR12/25/50/104 and DDR50).
- Cyclic Redundancy Check CRC7 for command and CRC16 for data integrity.
- Designed to work with I/O cards, Read-only cards and Read/Write cards.
- Supports Read wait Control, Suspend/Resume operation.

13.3.2 eMMC Interface Features

- eMMC v5 support
- Host clock rate variable between 0 and 200 MHz
- Supports HS400 mode.
- Up to 1600 Mbits per second data rate using 8 bit parallel data lines (High Speed DDR mode)
- Up to 3200 Mbits per second data rate using 8 bit parallel data lines (HS400 mode)
- Transfers the data in 1 bit, 4 bit and 8 bit modes.
- Cyclic Redundancy Check CRC7 for command and CRC16 for data integrity.
- Supports MMC Plus and MMC mobile.

13.3.3 Storage Interfaces overview

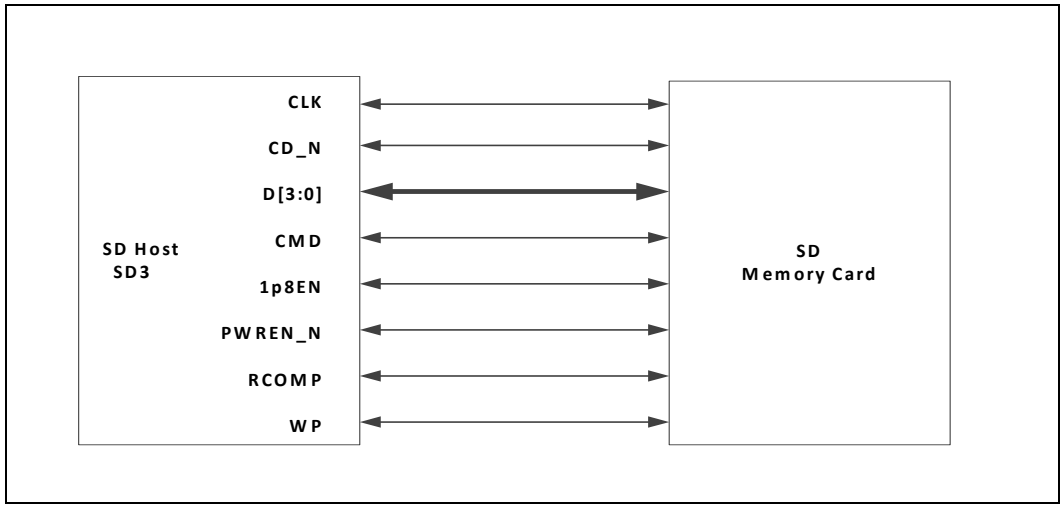
This section provides a very high level overview of the SD, SDIO, eMMC 5.0 specification.

13.3.3.1 SD 3.0 Bus Interface

The SD Card bus has a single master, single slaves (card), synchronous topology (refer to Figure 20). During initialization process commands are sent to the card, allowing the application to detect the card and assign logical addresses to the physical slot. All data communication in the Card Identification Mode uses the command line (CMD) only.

SD bus allows dynamic configuration of the number of data lines. After power up, by default, the SD Card will use only SD3_D[0] for data transfer. After initialization the host can change the bus width (number of active data lines). This feature allows easy trade off between hardware cost and system performance. Note that while DAT1-SD3_D[3:1] are not in use, the SoC will tri-state those signals.

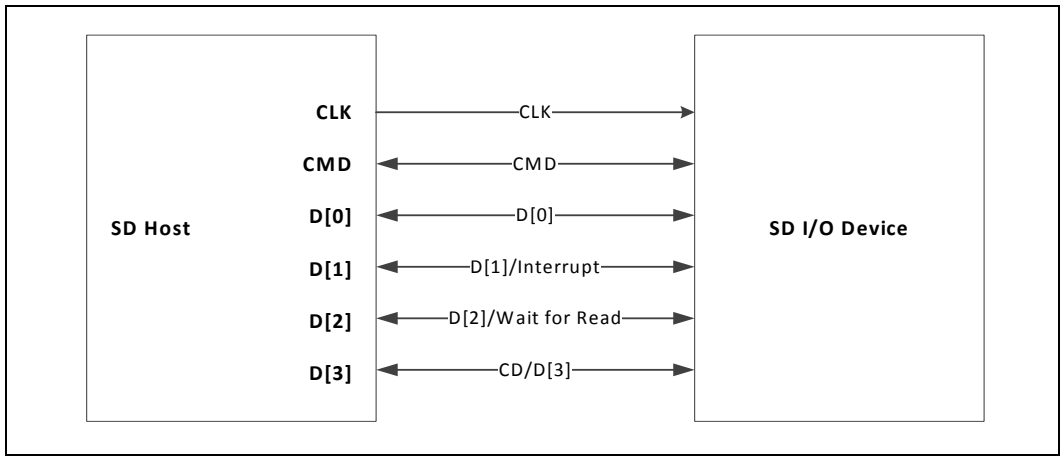
Figure 20. SD Memory Card Bus Topology



13.3.3.2 SDIO 3.0 Interface

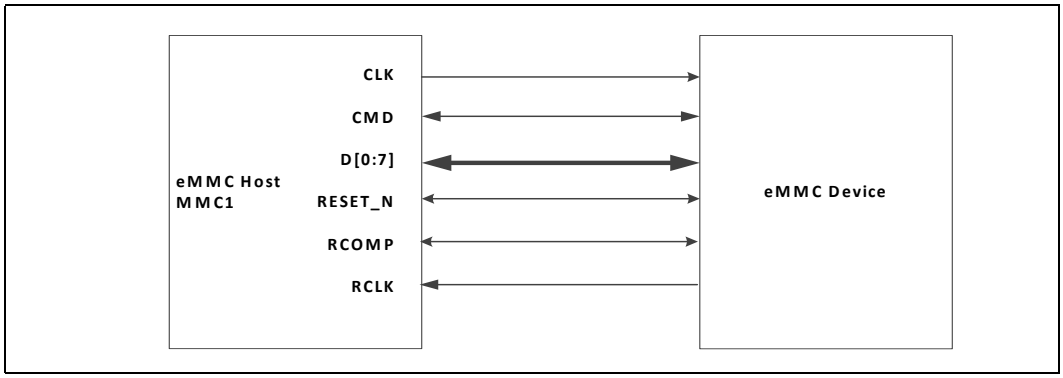
The SDIO interface is the very much like the SD card interface. The SoC supports one SDIO device.

Figure 21. SDIO Device Bus Topology



13.3.3.3 eMMC 5.0 Interface

Figure 22. eMMC Interface



The standard offers performance enhancement features, including HS400 support and has an interface bandwidth of 400 MByte/sec.

The command protocol is significantly improved with Packed Commands (the ability to group a series of commands in a single data transaction), Context ID (grouping different memory transactions under a single ID so the device can understand that they are related), and Data Tag (tagging specific write transactions so they can be prioritized and targeted to a memory region with higher performance and better reliability).

The v5.0 standard also adds provision for volatile data cache, which can greatly reduce the latency between data transactions to improve performance.

13.4 References

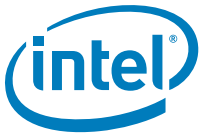
The controller is configured to comply with:

- SD Specification Part 01 Physical Layer Specification version 3.00, April 16, 2009
- SD Specification Part E1 SDIO Specification version 3.00, December 16, 2010
- SD Specification Part A2 SD Host Controller Standard Specification version 3.00, February 18, 2010
- SD Specification Part 03 security Specification version 1.01, April 15, 2001
- Embedded MultiMedia Card (eMMC) Product Standard v5.0, JESD84-A5

13.5 Register Map

For more information on SoC Storage registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§



14 USB Controller Interfaces

The USB Controller contains xHCI host controller that supports xHCI framework and USB1/2/3 specifications. And it has xDCI controller block for device only mode functionality. These 2 controllers will use an integrated mux to select between the 2 modes. All of this functionality is located in xDCI Controller.

14.1 SoC Supports

- Two (2) Super Speed Inter-Chip (SSIC) ports
- Three (3) Super Speed (SS) ports [Backward Compatible of USB 2.0 HS/FS/LS]
- One (1) Super Speed (SS) OTG port
- Two (2) High Speed Inter-Chip (HSIC) ports

Note: SoC can support the 4th SS port when OTG port is in Host mode.

14.2 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 76. USB SSIC Signals

Signal Name	Direction /Type	Description
USB_SSIC_RX_P/N[0,1]	I/O/ SSIC PHY	**Receiver serial data inputs:** High-speed serialized data inputs.
USB_SSIC_TX_P/N[0,1]	I/O/ SSIC PHY	**Transmitter serial data outputs:** High-Speed Serialized data outputs.
USB_SSIC_RCOMP_P/N	I / SSIC PHY	**Resistor Compensation:** An external resistor must be connected between this pin and package ground. See the platform design guidelines for more details.

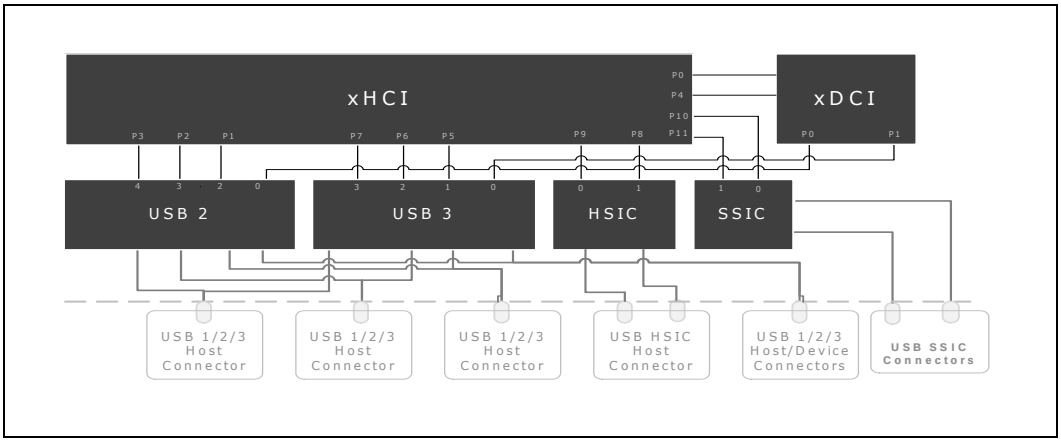
Table 77. USB Signals

Signal Name	Direction /Type	Description
USB3_TXP/N[0:3]	O USB3 PHY	**Transmitter serial data outputs:** High-Speed Serialized data outputs.
USB3_RXP/N[0:3]	I USB3 PHY	**Receiver serial data inputs:** High-speed serialized data inputs.
USB_RCOMP_P/N	I USB3 PHY	**Resistor Compensation:** An external resistor must be connected. See the platform design guidelines for details.
USB_DP/N[0:3]	I/O USB2 PHY	**USB2 Data:** High speed serialized data I/O.
USB_RCOMP	O USB2 PHY	**Resistor Compensation:** An external resistor must be connected. See the platform design guidelines for details.
USB_OTG_ID	I/O USB2 PHY	**OTG ID:** Pin out to detect the OTG ID.
USB_PLL_MON	O USB2 PHY	USB High Speed Observation
USB_VBUSSNS	I/O USB2 PHY	OTG Interface: VBUS_Sense

Table 78. HSIC Signals

Signal Name	Direction /Type	Description
USB_HSIC[0:1]_DATA	I/O HSIC Buffer	**HSIC Data**.
USB_HSIC[0:1]_STROBE	I/O HSIC Buffer	**HSIC Strobe**
USB_HSIC_RCOMP	I/O HSIC Buffer	**Resistor Compensation:** RCOMP for HSIC buffer. Resistor: 50Ohm +/-1% connected between USB_HSIC_RCOMP and ground.

Figure 23. xHCI Port Mapping



14.3 USB 3.0 xHCI (Extensible Host Controller Interface)

The xHCI compliant host controller can control up to 2 SSIC, 3 USB3.0 ports. USB3.0 being backward compatible to support USB2.0. It supports devices conforming to USB 1.x to 3.0 at bit rates up to 5 Gbps.

14.3.1 Features of USB 3.0 Host

The USB 3.0 Super Speed data interface is a four wire differential (TX and RX pairs) interface that supports simultaneous bi-directional data transmission. The interface supports a bit rate of 5 Gbps with a maximum theoretical data throughput over 3.2 Gbps due to 8b/10b symbol encoding scheme and protocol overhead (link flow control, packet framing and protocol overhead).

Low Frequency Periodic signaling (LFPS) is used to communicate initialization, training and power management information across a link that is in low power link state without using Super Speed signaling. This reduces power consumption.

14.3.1.1 USB SSIC Features

- Support for the SuperSpeed protocol only as defined in [USB 3.0]
- Optimized for Power, Area, Cost and EMI robustness.
- Support 2 ports of 1 lane each.

14.3.1.2 USB3.0 Features

- Supported by xHCI software host controller interface
- USB3 port disable
- Supports local dynamic clock gating and trunk clock gating

- Supports USB 3.0 LPM (U0, U1, U2, and U3) and also a SS Disabled low power state
- Support for USB3 Debug Device
- Supports IVCAM(USB PC Camera)

14.3.2 Features of USB HSIC

HSIC is a 2-signal (strobe and data) source synchronous serial interface for on board inter-chip USB communication. The interface uses 240 MHz DDR signaling to provide High-Speed 480 Mb/s USB transfers which are 100% host driver compatible with traditional USB cable connected topologies. Full Speed (FS) and Low Speed (LS) USB transfers are not directly supported by the HSIC interface.

Major feature and performance highlights are as follows:

- Supported by xHCI software host controller interface
- High-Speed 480 Mb/s data rate only
- Source-synchronous serial interface
- Power is only consumed when a transfer is in progress
- No Plug and Play support
- No hot plug removal/attach
- Signals driven at 1.2V standard LVCMOS levels
- Designed for low power applications
- Support for two host ports compliant to High Speed Inter-Chip Supplement (HSIC) to the USB 2.0 Specification. (USB 2.0)
- Clock request/ack mechanism is used

14.4 USB 3.0 xDCI (Extensible Device Controller Interface)

The xDCI compliant Device controller can control up to 1 USB3.0 OTG port. USB3 being backward compatible to support USB2.0. It supports devices conforming to USB 1.x to 3.0 at bit rates up to 5 Gbps.

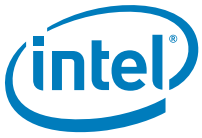
14.5 References

USB 3.0 Specification

USB 2.0 Specification (Includes High-Speed Inter-Chip USB Electrical Specification)

14.5.1 Host Controller Specifications

Extensible Host Controller Interface (xHCI) Specification for USB 3.0 version 1.0.



14.6 Register Map

For more information on USB Controller Interfaces registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.

§

15 Low Power Engine (LPE) for Audio (I^2S)

The Low Power Engine for Audio provides acceleration for common audio and voice functions. The voice and audio engine provides a mechanism for rendering audio and voice streams and tones from the operating system, applications to an audio or voice codec, and ultimately to the speaker, headphones, or Bluetooth headsets.

Audio streams in the SoC can be encoded and decoded by the Low Power Engine (LPE) in the Audio subsystem.

LPE Audio provides three external I^2S audio interfaces.

15.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 79. LPE Signals

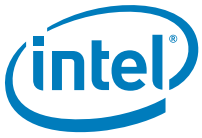
Signal Name	Direction /Type	Description
LPE_I2S[2:0]_CLK	I/O	Clock signal for I^2S
LPE_I2S[2:0]_FRM	I/O	Frame select signal for I^2S
LPE_I2S[2:0]_DATAIN	I/O	RX data for I^2S
LPE_I2S[2:0]_DATAOUT	I/O	TX data for I^2S

† All LPE signals are muxed and may be used by other functions.

15.2 Features

The LPE Audio Subsystem consists of the following:

- Integrated, power-efficient 32-bit architecture core with 24-bit audio processing instructions
- Core processing speeds up to 343 MHz
- Closely Coupled Memories (CCMs)
  - 80KB Instruction RAM



  - — 160KB Data RAM
  - — 48KB Instruction Cache
  - — 96KB Data Cache
- Very low-power consumption coupled with high-fidelity 24-bit audio
- Dual-issue, static, super-scalar VLIW processing engine
- Mode-less switching between 16-, 24-, and 64-bit dual-issue instructions
- Dual MACs which can operate with 32 x 16-bit and/or 24 x 24-bit operands
- Inter-Process Communication (IPC) mechanism to communicate with the SoC Processor Core including 4KB mailbox memory
- Flexible audio interfaces include three SSPs with I^2S port functionality for BI-directional audio transfers
  - — I^2S mode supports PCM payloads
  - — Frame counters for all I^2S ports
- High Performance DMA
  - — DMA IP to support multiple outstanding transactions
  - — Interleaved scatter-gather support for Audio DMA transfers
- Clock switching logic including new frequency increments
- External timer function with an always running clock.

The LPE core runs at a peak clock frequency of 343 MHz and has dedicated on-chip program and data memories and caches. The LPE core can access shared SRAM blocks, and external DRAM through OCP fabric. It communicates with audio peripherals using the audio sub-fabric, and employs Inter-Processor Communication (IPC) mechanism to communicate with the SoC Processor Core.

The Audio subsystem includes two OCP-based DMA engines. These DMA engines support single and multi-block transfers. They can be configured to transfer data between DRAM and audio CCMs or transfer data between CCMs and the audio peripheral interfaces

All these interfaces are peripherals in the Audio subsystem. LPE, LPE DMA, or the SoC processor core may access the peripherals during normal operation. The PMC may access all peripherals during specific tasks such as at boot time or during power state changes.A complete audio solution based on an internal audio processing engine which includes several I^2S-based output ports.

The audio core used is a dedicated audio DSP core designed specifically for audio processing (decoding, post-processing, mixing, etc.)

Note: LPE requires systems with more than 512 MB memory. This is required since the LPE firmware must reside at a stolen memory location on 512MB boundaries below 3 GiB. The LPE firmware itself is ~1MB, and is reserved by BIOS for LPE use.

15.2.1 Audio Capabilities

15.2.1.1 Audio Decode

The Audio core supports decoding of the following formats:

- MP3
- AAC-LC
- HE-AAC v1/2
- WMA9,10, PRO, Lossless, Voice
- MPEG layer 2
- RealAudio
- OggVorbis
- FLAC
- DD/DD+

15.2.1.2 Audio Encode

The Audio core supports encoding of the following formats:

- MP3
- AAC-LC
- WMA
- DD-2channel

15.3 Clocks

15.3.1 Clock Frequencies

Table 80 shows the clock frequency options for the Audio functional blocks.

Table 80. Clock Frequencies

Clock	Frequency	Notes
Audio core	343/250/200 MHz/100/ 50 MHz/2x Osc/Osc 50(RO)/100(RO)	Audio input clock trunk. CCU drives one of several frequencies as noted.
DMA 0	50/OSC	DMA clock
DMA1	50/OSC	DMA clock
Audio fabric clock	50/OSC	Fabric clock derived from audio core clock

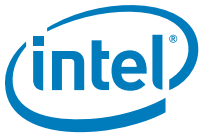


Table 80. Clock Frequencies

Clock	Frequency	Notes
SSP0 Clock	Fabric side: 50/OSC Link side: Up to 19.2 MHz	SSP0 clock domains
SSP1 Clock	Fabric side: 50/OSC Link side: Up to 19.2 MHz	SSP1 clock domains
SSP2 Clock	Fabric side: 50/OSC Link side: Up to 19.2 MHz	SSP2 clock domains

15.3.2 38.4 MHz Clock for LPE

38.4 MHz, the 2X OSC clock, is added to increase MIPS for low power MP3 mode. This frequency will be supplied by the clock doubler internal to the SoC's Clock Control Unit.

15.3.3 Calibrated Ring Osc (50/100 MHz) Clock for LPE:

A calibrated Ring Oscillator in the CCU_SUS provides a 50Mhz or an 100Mhz clock as another option for higher MIPS for low power MP3 mode. It is expected that this will be required to support decode of HE-AAC streams in the low power mode.

15.3.4 Cache and CCM Clocking

Data CCM, Data cache, Instruction CCM, and Instruction Cache run off of the LPE clock. These memories are in a single clock domain.

Note: All Data CCM and Instruction CCM run in the same clock domain.

15.4 SSP (I^2S)

The SoC audio subsystem consists of the LPE Audio Engine and three Synchronous Serial Protocol (SSP) ports. These ports are used in PCM mode and enable simultaneous support of voice and audio streams over I^2S. The SoC audio subsystem also includes two DMA controllers dedicated to the LPE. The LPE DMA controllers are used for transferring data between external memory and CCMs, between CCMs and the SSP ports, and between CCMs. All peripheral ports can operate simultaneously.

15.4.1 Introduction

The Enhanced SSP Serial Ports are full-duplex synchronous serial interfaces. They can connect to a variety of external analog-to-digital (A/D) converters, audio, and telecommunication codecs, and many other devices which use serial protocols for transferring data. Formats supported include National* Microwire, Texas Instruments* Synchronous Serial Protocol (SSP), Motorola* Serial Peripheral Interface (SPI) protocol and a flexible Programmable Serial Port protocol (PSP).

The Enhanced SSPs operate in master mode (the attached peripheral functions as a slave) or slave mode (the attached peripheral functions as a master), and support serial bit rates from 0 to 25 Mbps, dependent on the input clock. Serial data formats range from 4 to 32-bits in length. Two on-chip register blocks function as independent FIFOs for transmit and receive data.

FIFOs may be loaded or emptied by the system processor using single transfers or DMA burst transfers of up to the FIFO depth. Each 32-bit word from the bus fills one entry in a FIFO using the lower significant bits of a 32-bit word.

15.4.2 SSP Features

The SSP port features are:

- Inter-IC Sound (I^2S) protocols, are supported by programming the Programmable Serial Protocol (PSP).
- One FIFO for transmit data (TXFIFO) and a second, independent, FIFO for receive data (RXFIFO), where each FIFO is 16 samples deep x 32 bits wide
- Data sample sizes from 8, 16, 18, or 32 bits
- 12.5 Mbps maximum serial bit-rate in both modes: master and slave.
- Clock master or slave mode operations
- Receive-without-transmit operation
- Network mode with up to eight time slots for PSP formats, and independent transmit/receive in any/all/none of the time slots.
- After updating SSP configuration, for example active slot count, the SSP will need to be disabled and enabled again. In other words, a SSP will not function correctly if a user changes the configuration setting on the fly.

§

16 Intel® Trusted Execution Engine (Intel® TXE)

This chapter describes the security components and capabilities. The security system contains an Intel® TXE and additional hardware security feature that enable a secure and robust platform.

16.1 Features

16.1.1 Security Feature

The Intel® TXE in the SoC is responsible for supporting and handling security related features.

Intel TXE features:

- 32-bit RISC processor
- 256KB Data/Code RAM accessible only to the Intel® TXE
- 128KB On Chip Mask ROM for storage of Intel® TXE code
- Inter-Processor Communication for message passing between the Host CPU and Intel® TXE
- 64 byte input and output command buffers
- 256 byte shared payload (enables 2048-bit keys to be exchanged as part of the command)
- Multiple context DMA engine to transfer data between Host CPU address domain (System memory) and the Intel® TXE; programmable by the Intel® TXE CPU only.
- Secure I^2C interface to NFC using master I^2C block integrated into the Intel TXE - IP. Secure GPIOs to support input alert and two GP Outputs.

16.1.1.1 HW Accelerators

- DES/3DES (ECB, CBC) – 128b ABA key for 3DES Key Ladder Operations
- Three AES engines - Two fast -128 and one slow- 128/256
- Exponentiation Acceleration Unit (EAU) for modular exponentiation, modular reduction, large number addition, subtraction, and multiplication
- SHA1, SHA256/384/512, MD5

16.1.1.2 FW Utilities and Ciphers

- RSA (with EAU acceleration)
- Flash Write Enable/Disable

- Comprehensive IPC Command Set
- Chip Unique Key encryption key wrapping of other platform keys (Flash)

16.1.1.3 Downloadable FW Utilities and Ciphers

- Integrated Theft Deterrence Technology - Intel® Anti-Theft Technology (Intel® AT)
- One Time Programmable (OTP)
- Firmware TPM (fTPM) measured boot

16.1.2 The TXE interaction with NFC

— The NFC device requests attention from the TXE from GPIO_ALERT pin to a SoC input interrupt pin (GPIO_SUS[8] pin).

— The GPIO block sends the pin value to TXE over a dedicated wire.

— The wire is connected to the TXE clock request mechanism in order to get a clock for sampling the wire. The TXE bridge includes a configuration register which includes an enable bit to qualify the clock request (which allows masking the clock request, in case the GPIO_SUS[8] is not used by NFC), and a polarity bit (which allows selecting whether the a clock request would be set on a high or low value in the wire).)

— The same qualified & polarity configured clock request input is also sent to PMU. In S0ix PMU uses it as a wake request.

— When a clock is available, the wire value is updated to an ICR (SICR31) in TXE bridge.

— TXE Bridge configuration register also includes two bits that allow detection of falling and/or rising edge on the alert pin. They cause an ISR (SISR[31]) to be set. When both ISR and IER bits for the alert are set an interrupt is generated.

— When the TXE is interrupted it parses the interrupt status registers in the TXE Bridge and figures the cause is the NFC device.

— TXE clears the Bridge ISR and sets configuration to detect the next edge on the alert pin.

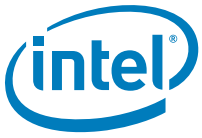
— In order to use the I2C master, the TXE sets an I2C clock request register in the Bridge.

— The firmware then uses the I2C master to communicate with the NFC device. The firmware configures the I2C master to read up to 33Bytes of data (up to 36 bytes are supported by HW for read/write).

— When the I2C read is completed, the firmware is interrupted. The TXE may then read the data/status and clear the interrupt.

— The firmware repeats read/write sequence’s as many times as it needs.

— When firmware is done with the I2C master, it must poll the controller to check that the I2C bus is idle before writing to the register to remove the I2C clock request, and before any reset of the I2C controller or power gating sequence.



Shutting off clock or I2C master before the completion all activity on the bus will hang the I2C device.

§

17 Intel® Sensor Hub

This chapter describes Intel® Sensor Hub

17.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 81. ISH Signals

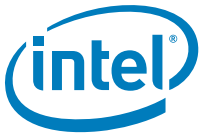
Signal Name	Direction	Description
ISH_I2C1_CLK	I/O	**Clock Lane:** ISH input clock.
ISH_I2C1_SDA	I/O	**Data Lane:** ISH Data Lane
ISH_GPIO	I/O	ISH GPIO

17.2 Features

- 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 gating of the ISH together with the ability to turn sensors off under control of host SW
- The ability to operate independently when the host platform is in a low power state(S0-S0i3)
- Power saving features
- Clock gating and power gating of functional blocks depending on current workloads

17.2.1 Hardware Overview

- Minute IA microprocessor
- 384KB on chip Data/Code SRAM accessible only to the ISH
- 8KB on chip ROM for ISH boot code



- Inter-Processor Communication for message passing between the Host CPU and Intel® ISH
  - — Single Command/Doorbell DWORD register and 32 DWORD data registers each direction.
- Inter-Processor Communication for message passing between the Intel® ISH and Intel® TXE for ISH FW load
  - — Single Command/Doorbell DWORD register and 32 DWORD data registers each direction.
- Inter-Processor Communication for message passing between the Intel® ISH and PMC for ISH power management and ISH TXE communication assistance by PMC
  - — Single Command/Doorbell DWORD register each direction.
- DMA engine to transfer data between Host CPU address domain (System memory) and the Intel® ISH; programmable by the Intel® ISH CPU only.
- Two I2C interfaces and up to 15 GPIO lines for connecting sensors to ISH.

§

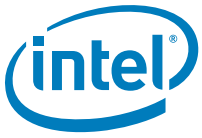
18 Serial IO (SIO) Overview

The Serial I/O (SIO) is a collection of hardware blocks that implement simple but key serial I/O interfaces for platform usage. These hardware blocks include:

- "SIO - I^2C Interface"
- "SIO - High Speed UART"
- "SIO - Serial Peripheral Interface (SPI)"
- "SIO - Pulse Width Modulation (PWM)"

18.1 Register Map

For more information on SIO registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.



18.2 SIO - Serial Peripheral Interface (SPI)

The Serial I/O implements three SPI controllers that supports master mode.

18.2.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 82. SPI Interface Signals

Signal Name	Direction /Type	Description
SPI[1,2,3]_CLK	O GPIO	**SPI Clock:** When the bus is idle, the owner will drive the clock signal low.
SPI[1,2,3]_CS[0]_N	O GPIO	**SPI Chip Select 0:** Used as the SPI Chip select 0.
SPI[1,2,3]_CS[1]_N	O GPIO	**SPI Chip Select 1:** Used as the SPI Chip select 1.
SPI[1,2,3]_MISO	I GPIO	**SPI Master IN Slave OUT:** Data input pin for the SoC.
SPI[1,2,3]_MOSI	O GPIO	**SPI Master OUT Slave IN:** Data output pin for the SoC. Operates as a second data input pin for the SoC when in Single Input, Dual Output Fast Read mode.

18.2.2 Features

The following is a list of SPI features:

- Single interrupt line
  - — Could be assigned to interrupt PCI INT [A] or ACPISIO INT[1]
- Configurable frame format, clock polarity and clock phase
- supporting three SPI peripherals only
- Two Chip selects are supported for each of the 3 SPI controllers.
- Supports master mode only
- Receive and transit buffers are both 256x32 Bits
  - — The receive buffer has only 1 water mark
  - — The transmit buffer has 2 water marks

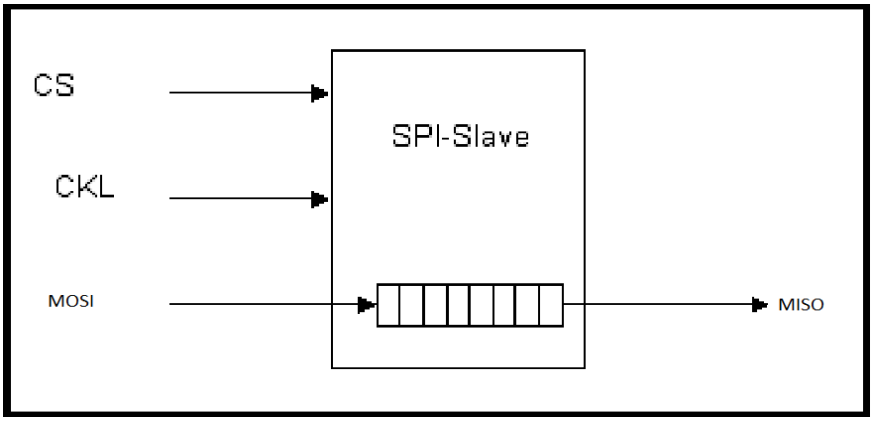
- Supports up to 20 Mbps

18.2.2.1 General overview

The Serial Peripheral Interface is used primarily for a synchronous serial communication of host processor and peripherals.

In the standard configuration for a slave device, two control and two data lines are used. The data output MISO serves on the one hand the reading back of data, offers however also the possibility to cascade several devices. The data output of the preceding device then forms the data input for the next IC.

Figure 24. SPI Slave



There is a MASTER and a SLAVE mode. The MASTER device provides the clock signal and determines the state of the chip select lines, i.e. it activates the SLAVE it wants to communicate with. CS and CKL are therefore outputs. The SLAVE device receives the clock and chip select from the MASTER, CS and CKL are therefore inputs. This means there is one master, while the number of slaves is only limited by the number of chip selects.

A SPI device can be a simple shift register up to an independent subsystem. The basic principle of a shift register is always present. Command codes as well as data values are serially transferred, pumped into a shift register and are then internally available for parallel processing.

The SPI requires two control lines (CS and CLK) and two data lines MOSI (Master-Out-Slave-In) and MISO (Master-In-Slave-Out).

18.2.2.2 Data and Control lines for SPI

With CS (Chip-Select) the corresponding peripheral device is selected. This pin is mostly active-low. In the un-selected state the MISO lines are hi-Z and therefore inactive. The master decides with which peripheral device it wants to communicate.

The clock line CLK is brought to the device whether it is selected or not. The clock serves as synchronization of the data communication. The majority of SPI devices provide these four lines. Sometimes it happens that MOSI and MISO are multiplexed.

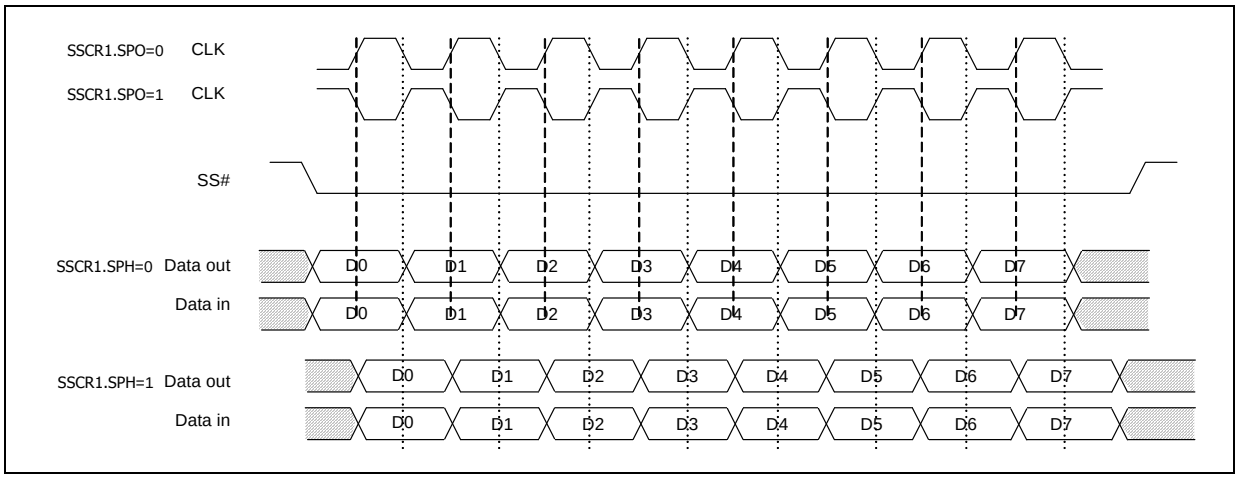
18.2.2.3 SPI Configuration: Clock Phase and Polarity

SPI clock phase and clock polarity overview.

- The SSCR1.SPO polarity setting bit determines whether the serial transfer occurs on the rising edge of the clock or the falling edge of the clock.
  - When SSCR1.SPO = 0, the inactive or idle state of SPI1_CLK is low.
  - When SSCR1.SPO = 1, the inactive or idle state of SPI1_CLK is high.
- The SSCR1.SPH phase setting bit selects the relationship of the serial clock with the slave select signal.
  - When SSCR1.SPH = 0, SPI1_CLK is inactive until one cycle after the start of a frame and active until 1/2 cycle after the end of a frame.
  - When SSCR1.SPH = 1, SPI1_CLK is inactive until 1/2 cycle after the start of a frame and active until one cycle after the end of a frame.

Below figure shows an 8-bit data transfer with different phase and polarity settings.

Figure 25. Clock Phase and Polarity

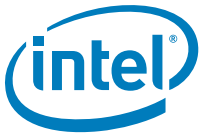


In a single frame transfer, the SPI controller supports all four possible combinations for the serial clock phase and polarity.

The combinations of polarity and phases are referred to as modes which are commonly numbered according to the following convention, with SSCR1.SPO as the high order bit and SSCR1.SPH as the low order bit.

Table 83. SPI Modes

Mode	SSCR1.SPO	SSCR1.SPH
0	0	0
1	0	1
2	1	0
3	1	1



18.3 SIO - I²C Interface

The SoC supports 7 instances of I²C controller. Both 7-bit and 10-bit addressing modes are supported. These controllers operate in master mode only.

18.3.1 Signal Descriptions

I²C is a two-wire bus for inter-IC communication. Data and clock signals carry information between the connected devices. The following is the I²C Interface. The SoC supports 7 I²C interfaces for general purpose to control external devices. The I²C signals are muxed over GPIOs.

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

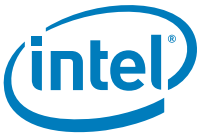
Table 84. I²C[6:0] Signals

Signal Name	Direction /Type	Description
I2C[6:0]_DATA	I/O/ GPIOMV, MS, I2C	**I²C Serial Data** *These signals are muxed and may be used by other functions.*
I2C[6:0]_CLK	I/O/ GPIOMV, MS, I2C	**I²C Serial Clock** *These signals are muxed and may be used by other functions.*

18.3.2 NFC I²C Interface Signals

Table 85. NFC I²C Interface Signals

Signal Name	Direction / Type	Description
NFC_I2C_DATA	I/O/ GPIOMV, MS, I2C	**NFC I²C Serial Data** *These signals are muxed and may be used by other functions.*
NFC_I2C_CLK	I/O/ GPIOMV, MS, I2C	**NFC I²C Serial Clock** *These signals are muxed and may be used by other functions.*
GPIO_ALERT	I/O/ GPIOMV, MS	**ALERT pin for NFC** *These signals are muxed and may be used by other functions.*



18.3.3 Features

18.3.3.1 I2C Protocol

The I2C bus is a two-wire serial interface, consisting of a serial data line and a serial clock. These wires carry information between the devices connected to the bus. Each device is recognized by a unique address and can operate as either a "transmitter" or "receiver," depending on the function of the device. Devices are considered slaves when performing data transfers, as the SoC will always be a Master. A master is a device which initiates a data transfer on the bus and generates the clock signals to permit that transfer. At that time, any device addressed is considered a slave.

- The SoC is always the I2C master; and it supports multi-master mode.
- The SoC can support clock stretching by slave devices.
- The I2Cx_DATA line is a bidirectional signal and changes only while the I2Cx_CLK line is low, except for STOP, START, and RESTART conditions.
- The output drivers are open-drain or open-collector to perform wire-AND functions on the bus.
- The maximum number of devices on the bus is limited by the maximum capacitance specification of 400 pF
- Refer to Chapter 20, "Electrical Specifications" for details.
- Data is transmitted in byte packages.

18.3.3.2 I2C Modes of Operation

The I2C module can operate in the following modes:

- Standard mode (with a bit rate up to 100 Kb/s)
- Fast mode (with a bit rate up to 400 Kb/s)
- Fast Mode plus mode (with a bit rate up to 1 Mb/s)
- High-speed mode (with a bit rate up to 1.7 Mb/s)

The I2C can communicate with devices only using these modes as long as they are attached to the bus. Additionally, high speed mode, fast mode plus and fast mode devices are downward compatible.

- High-speed mode devices can communicate with fast mode and standard mode devices in a mixed speed bus system.
- Fast mode devices can communicate with standard mode devices in a 0–100 Kb/s I2C bus system.

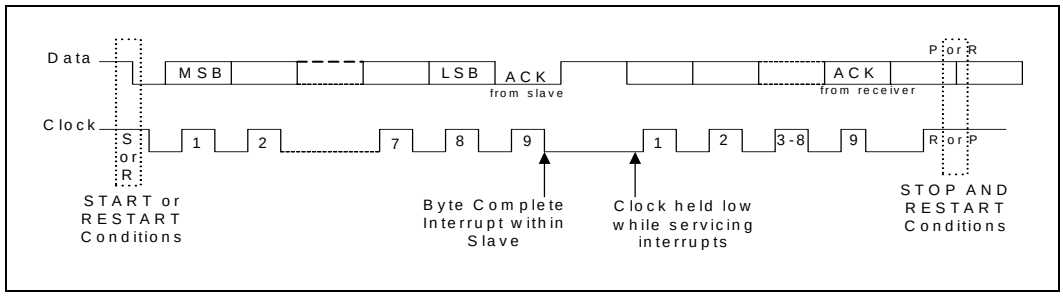
However, according to the I2C specification, standard mode devices are not upward compatible and should not be incorporated in a fast-mode I2C bus system since they cannot follow the higher transfer rate and unpredictable states would occur.

Refer to Table 1 for more information on the I2C interface speed for different Sku's.

18.3.3.3 Functional Description

- The I2C master is responsible for generating the clock and controlling the transfer of data.
- The slave is responsible for either transmitting or receiving data to/from the master.
- The acknowledgement of data is sent by the device that is receiving data, which can be either a master or a slave.
- Each slave has a unique address that is determined by the system designer:
  - When a master wants to communicate with a slave, the master transmits a START/RESTART condition that is then followed by the slave's address and a control bit (R/W), to determine if the master wants to transmit data or receive data from the slave.
  - The slave then sends an acknowledge (ACK) pulse after the address.
- If the master (master-transmitter) is writing to the slave (slave-receiver)
  - The receiver gets one byte of data.
  - This transaction continues until the master terminates the transmission with a STOP condition.
- If the master is reading from a slave (master-receiver)
  - The slave transmits (slave-transmitter) a byte of data to the master, and the master then acknowledges the transaction with the ACK pulse.
  - This transaction continues until the master terminates the transmission by not acknowledging (NACK) the transaction after the last byte is received, and then the master issues a STOP condition or addresses another slave after issuing a RESTART condition. This behavior is illustrated in below figure.

Figure 26. Data Transfer on the I2C Bus



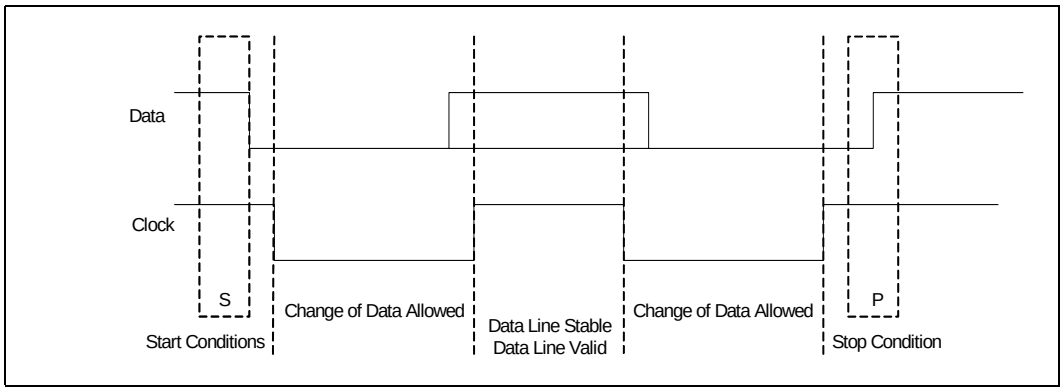
18.3.3.3.1 START and STOP Conditions

When the bus is idle, both the clock and data signals are pulled high through external pull-up resistors on the bus.

When the master wants to start a transmission on the bus, the master issues a START condition.

- This is defined to be a high-to-low transition of the data signal while the clock is high.
- When the master wants to terminate the transmission, the master issues a STOP condition. This is defined to be a low-to-high transition of the data line while the clock is high. Figure 27 shows the timing of the START and STOP conditions.
- When data is being transmitted on the bus, the data line must be stable when the clock is high.

Figure 27. START and STOP Conditions



The signal transitions for the START/STOP conditions, as depicted above, reflect those observed at the output of the master driving the I^2C bus. Care should be taken when observing the data/clock signals at the input of the slave(s), because unequal line delays may result in an incorrect data/clock timing relationship.

18.4 NFC I^2C

NFC device requires 1.8V I/Os.

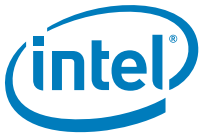
For more information refer to "The TXE interaction with NFC"

18.4.1 References

I^2C-Bus Specification and User Manual, Revision 03: http://ics.nxp.com/support/documents/interface/pdf/i2c.bus.specification.pdf

18.4.2 Register Map

Refer to Chapter 4, "Register Access Methods" and Chapter 5, "Mapping Address Spaces" for additional information.



18.5 SIO - High Speed UART

The SoC implements two instances of high speed UART controller that support baud rates between 300 and 3686400. Hardware flow control is also supported.

18.5.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 86. UART 1 Interface Signals

Signal Name	Direction / Type	Description
UART1_DATAIN	I/O/ GPIOMV, MS	**High-speed UART receive data input**: *This signal is muxed and may be used by other functions.*
UART1_DATAOUT	I/O/ GPIOMV, MS	**High-speed UART transmit data:** *This signal is muxed and may be used by other functions.*
UART1_RTS_N	I/O/ GPIOMV, MS	**High-speed UART request to send:** *This signal is muxed and may be used by other functions.*
UART1_CTS_N	I/O/ GPIOMV, MS	**High-speed UART clear to send:** *This signal is muxed and may be used by other functions.*

Table 87. UART 2 Interface Signals (Sheet 1 of 2)

Signal Name	Direction / Type	Description
UART2_DATAIN	I/O/ GPIOMV, MS	**High-speed UART receive data input:** *This signal is muxed and may be used by other functions.*

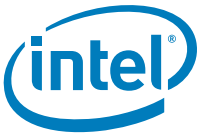


Table 87. UART 2 Interface Signals (Sheet 2 of 2)

Signal Name	Direction / Type	Description
UART2_DATAOUT	I/O/ GPIOMV, MS	**High-speed UART transmit data:** *This signal is muxed and may be used by other functions.*
UART2_RTS_N	I/O/ GPIOMV, MS	**High-speed UART request to send:** *This signal is muxed and may be used by other functions.*
UART2_CTS_N	I/O/ GPIOMV, MS	**High-speed UART clear to send:** *This signal is muxed and may be used by other functions.*

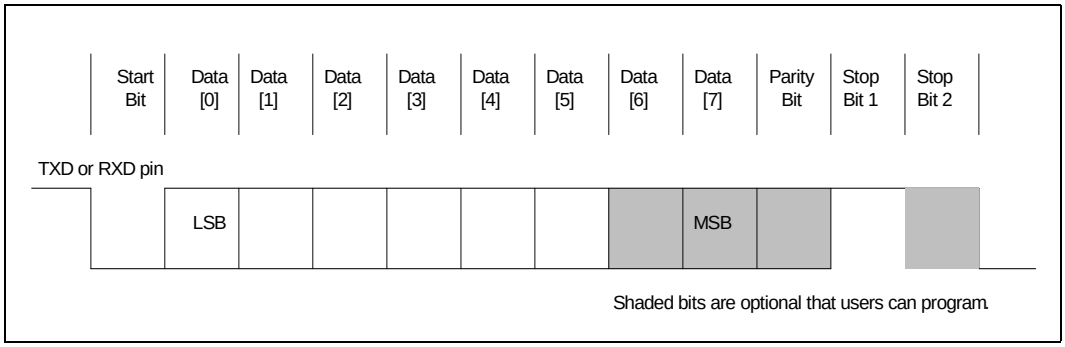
18.5.2 Features

18.5.2.1 UART Function

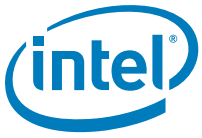
The UART transmits and receives data in bit frames as shown in Figure 29.

- Each data frame is between 7 and 12 bits long, depending on the size of data programmed and if parity and stop bits are enabled.
- The frame begins with a start bit that is represented by a high-to-low transition.
- Next, 5 to 8 bits of data are transmitted, beginning with the least significant bit. An optional parity bit follows, which is set if even parity is enabled and an odd number of ones exist within the data byte; or, if odd parity is enabled and the data byte contains an even number of ones.
- The data frame ends with one, one-and-one-half, or two stop bits (as programmed by users), which is represented by one or two successive bit periods of a logic one.

Figure 29. UART Data Transfer Flow



Each UART has a Transmit FIFO and a Receive FIFO and each holds 64 characters of data. There are two separate methods for moving data into/out of the FIFOs—Interrupts and Polling.



18.5.2.2 Clock and Reset

The BAUD rate generates from base frequency of 50 MHz.

18.5.2.3 Baud Rate Generator

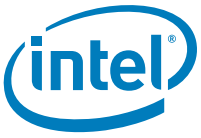
The baud rates for the UARTs are generated with from the base frequency (Fbase) indicated in Table 88 by programming the DLH and DLL registers as divisor. The hexadecimal value of the divisor is (IER_DLH[7:0]<<8) | RBR_THR_DLL[7:0].

Fbase 44236800 Hz can be achieved by programming the DDS Multiplier as 44,236,800 (in decimal), and DDS Divisor as the system clock frequency in Hz. (50,000,000 in decimal when the system clock frequency is 50 MHz.)

The output baud rate 3686400 is equal to the base frequency divided by thirteen times the value of the divisor, as follows: baud rate = (Fbase) / (13 * divisor). The output baud rate for all other baud rates is equal to the base frequency divided by sixteen times the value of the divisor, as follows: baud rate = (Fbase) / (16 * divisor).

Table 88. Baud Rates Achievable with Different DLAB Settings

DLH,DLL Divisor	DLH,DLL Divisor Hexadecimal	Baud Rate
Fbase 1: 47923200 Hz		
1	0001	3686400
Fbase 2: 44236800 Hz		
1	0001	2764800
3	0003	921600
6	0006	460800
9	0009	307200
12	000C	230400
15	000F	184320
18	0012	153600
24	0018	115200
48	0030	57600
72	0048	38400
144	0090	19200
288	0120	9600
384	0180	7200
576	0240	4800
768	0300	3600
1152	0480	2400
1536	0600	1800
2304	0900	1200
4608	1200	600
9216	2400	300



18.5.3 Use

Each UART has a transmit FIFO and a receive FIFO, each FIFO holding 64 characters of data. Three separate methods move data into and out of the FIFOs: interrupts, DMA, and polled.

18.5.3.1 DMA Mode Operation

18.5.3.1.1 Receiver DMA

The data transfer from the HSUART to host memory is controlled by the DMA write channel. To configure the channel in write mode, channel direction in the channel control register needs to be programmed to "1". The software need to program the descriptor start address register, descriptor transfer size register, and descriptor control register before starting the channel using the channel active bit in the channel control register.

18.5.3.1.2 Transmit DMA

The data transfer from host memory to HSUART is controlled by DMA read channel. To configure the channel in read mode, channel direction in the channel control register needs to be programmed to "0". The software need to program the descriptor start address register, descriptor transfer size register, and descriptor control register before starting the channel using the channel active bit in the channel control register.

18.5.3.1.3 Removing Trailing Bytes in DMA Mode

When the number of entries in the Receive FIFO is less than its trigger level, and no additional data is received, the remaining bytes are called Trailing bytes. These are DMAed out by the DMA as it has visibility into the FIFO Occupancy register.

18.5.3.2 FIFO Polled-Mode Operation

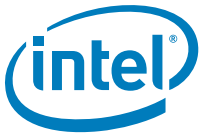
With the FIFOs enabled (IIR_FCR.IID0_FIFOE bit set to 1), clearing IER_DLH[7] and IER_DLH[4:0] puts the serial port in the FIFO Polled Operation mode. Because the receiver and the transmitter are controlled separately, either one or both can be in Polled Operation mode. In this mode, software checks Receiver and Transmitter status using the Line Status Register (LSR). The processor polls the following bits for Receive and Transmit Data Service.

18.5.3.2.1 Receive Data Service

The processor checks data ready (LSR.DR) bit which is set when 1 or more bytes remains in the Receive FIFO or Receive Buffer Register (RBR_THR_DLL).

18.5.3.2.2 Transmit Data Service

The processor checks transmit data request LSR.THRE bit, which is set when the transmitter needs data.



The processor can also check transmitter empty LSR.TEMT, which is set when the Transmit FIFO or Holding register is empty.

18.5.3.2.3 Autoflow Control

Autoflow Control uses Clear-to-Send (nCTS) and Request-to-Send (nRTS) signals to automatically control the flow of data between the UART and external modem. When autoflow is enabled, the remote device is not allowed to send data unless the UART asserts nRTS low. If the UART de-asserts nRTS while the remote device is sending data, the remote device is allowed to send one additional byte after nRTS is deasserted. An overflow could occur if the remote device violates this rule. Likewise, the UART is not allowed to transmit data unless the remote device asserts nCTS low. This feature increases system efficiency and eliminates the possibility of a Receive FIFO Overflow error due to long interrupt latency.

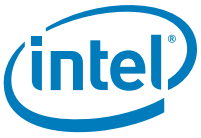
Autoflow mode can be used in two ways: Full autoflow, automating both nCTS and nRTS, and half autoflow, automating only nCTS. Full Autoflow is enabled by writing a 1 to bits 1 and 5 of the Modem Control Register (MCR). Auto-nCTS-Only mode is enabled by writing a 1 to bit 5 and a 0 to bit 1 of the MCR register.

18.5.3.2.4 RTS (UART Output)

When in full autoflow mode, nRTS is asserted when the UART FIFO is ready to receive data from the remote transmitter. This occurs when the amount of data in the Receive FIFO is below the programmable threshold value. When the amount of data in the Receive FIFO reaches the programmable threshold, nRTS is de-asserted. It will be asserted once again when enough bytes are removed from the FIFO to lower the data level below the threshold.

18.5.3.2.5 CTS (UART Input)

When in Full or Half-Autoflow mode, nCTS is asserted by the remote receiver when the receiver is ready to receive data from the UART. The UART checks nCTS before sending the next byte of data and will not transmit the byte until nCTS is low. If nCTS goes high while the transfer of a byte is in progress, the transmitter will complete this byte.



18.6 SIO - Pulse Width Modulation (PWM)

The Pulse Width Modulation block allows control the frequency and duty cycle of an output signal. The SoC has 2 instances of the PWM interface.

18.6.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

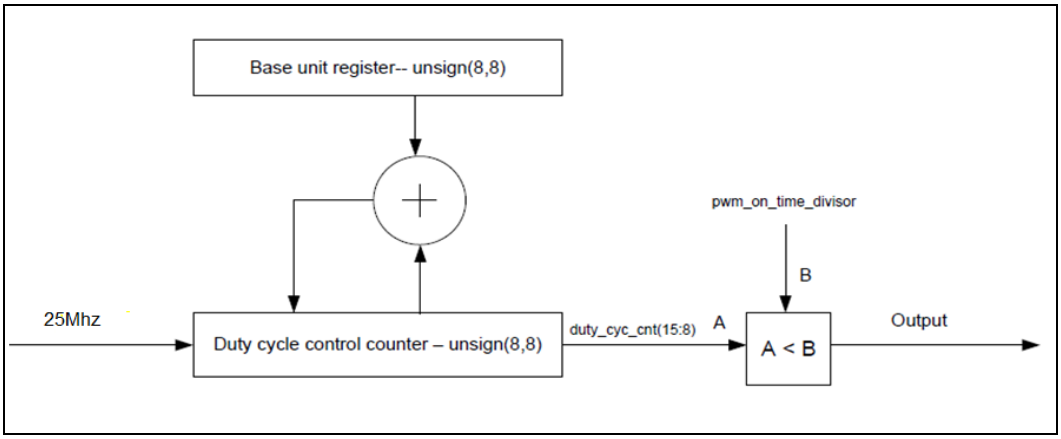
Figure 30. PWM Signals

Signal Name	Direction / Type	Description
PWM[0]	I/O/ GPIOMV, MS	Pulse Width Modulation output 0.
PWM[1]	I/O/ GPIOMV, MS	Pulse Width Modulation output 1.

18.6.2 Features

The software controls the PWM block by updating the PWMCTRL register and setting the PWMCTRL.PWM_SW_UPDATE bit whenever a change in frequency or duty cycle of the PWM output signal is required. The PWM block applies the new settings at the start of the next output cycle and resets the PWMCTRL.PWM_SW_UPDATE bit. The SoC uses 25 MHz for the counter. See Figure 31 for PWM block diagram:

Figure 31. PWM Block Diagram



There are two controls of the PWM output:

- Frequency is controlled by the PWMCTRL.PWM_BASE_UNIT bits. The PWMCTRL.PWM_BASE_UNIT value is added to a 16 bit counter every clock cycle and the counter roll-over marks the start of a new cycle.
- **Duty cycle** is controlled by the PWMCTRL.PWM_ON_TIME_DIVISOR setting (0 to 255). When the counter rolls-over it is reset and a new cycle starts with the output signal being 0, once the counter reaches the PWMCTRL.PWM_ON_TIME_DIVISOR value the output toggles to 1 and stays high until the counter rolls over.

The PWM block is clocked by the 25 MHz oscillator clock. The output frequency can be estimated with the equation:

- Target frequency = 25MHz * base_unit value/256

  Note that the larger the value of base_unit, the larger the error that the PWM output frequency will have with respect to the equation above. For example any Base_unit_value > 128 will result in 12.5 MHz max frequency. Any value between 86 and 128 will result in 8.33 MHz output frequency. And accordingly the larger the base_unit value the smaller duty cycle resolution. Maximum duty cycle resolution is 8 bits.

Table 89 illustrates the output frequency and duty-cycle resolution for different settings of the base_unit_value (when using 25 MHz oscillator clock).

Table 89. Example PWM Output Frequency and Resolution

Target Frequency	Base Unit Value	CLK Cycle Count	Duty Cycle Resolution
12.5 MHz	>=128	1	no resolution
1.07 MHz	11	23	<8 bit resolution
488 kHz	5	51	<8 bit resolution

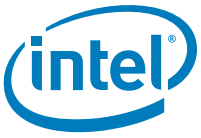
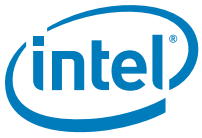


Table 89. Example PWM Output Frequency and Resolution

Target Frequency	Base Unit Value	CLK Cycle Count	Duty Cycle Resolution
97.6 kHz	1	256	8 resolution
48.8 kHz	0.5	Theoretically 512 but only 255 available since On Time Divisor is only 8b	>8bit
0	0	0	Flat 0 output

§



19 Platform Controller Unit (PCU) Overview

The Platform Controller Unit (PCU) is a collection of HW blocks that are critical for implementing a Windows* compatible platform. These HW blocks include:

- "PCU - Power Management Controller (PMC)"
- "PCU - Fast Serial Peripheral Interface (SPI)"
  - For boot FW and system configuration data Flash storage
- "PCU - Universal Asynchronous Receiver/Transmitter (UART)"
- "PCU - Intel Legacy Block (iLB) Overview"

The PCU also implements some high level configuration features for BIOS/EFI boot.

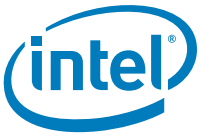
19.1 Features

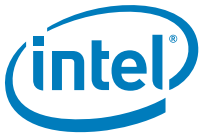
The key features of the individual blocks are as follows:

- Universal Asynchronous Receiver/Transmitter (UART)
  - 16550 controller compliant
  - Reduced Signal Count: TX and RX only
  - COM1 interface
- Fast Serial Peripheral Interface (FST_SPI)
  - For SPI Flash, of up to 16MB size per chip select is supported. No other SPI peripherals are supported.
  - Stores boot FW and system configuration data
  - Supports frequencies of 20 MHz, 33MHz and 50MHz.
- Power Management Controller (PMC)
  - Controls many of the power management features present in the SoC.
- Intel Legacy Block (iLB)
  - Supports legacy PC platform features
  - Sub-blocks include LPC, GPIO, 8259 PIC, IO-APIC, 8254 timers, HPET timers and the RTC.

19.2 Register Map

For more information on PCU registers refer Cherry Trail SoC External Design Specification (EDS) (Volume 2 of 2) Doc ID 543698.





19.3 PCU - Power Management Controller (PMC)

The Power Management Controller (PMC) controls many of the power management features present in the SoC.

19.3.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 90. PMC Signals (Sheet 1 of 2)

Signal Name	Direction /Type	Description
PMC_ACPRESENT	I/O/ GPIOMV, MS	**AC Present:** This input pin indicates when the platform is plugged into AC power.
PMC_BATLOW_N	I/O/ GPIOMV, MS	**Battery Low:** An input from the battery to indicate that there is insufficient power to boot the system. Assertion will prevent wake from the S4/S5 state. This signal can also be enabled to cause an SMI_N when asserted. In desktop configurations without a battery, this signal should be tied high to V1P8_S5.
PMC_CORE_PWROK	I/GPIOMV, MS	**Core Power OK:** When asserted, this signal is an indication to the SoC that all of its core power rails have been stable for 10 ms. It can be driven asynchronously. When it is negated, the SoC asserts PMC_PLTRST_N. **NOTE:** It is required that the power rails associated with PCI Express (typically the 3.3V, 5V, and 12V core well rails) have been valid for 99 ms prior to PMC_CORE_PWROK assertion in order to comply with the 100 ms T_{PVPERL} PCI Express 2.0 specification on PMC_PLTRST_N deassertion. **NOTE:** PMC_CORE_PWROK must not glitch, even if PMC_RSMRST_N is low.
PMC_PLTRST_N	I/O/ GPIOMV, MS	**Platform Reset:** The SoC asserts this signal to reset devices on the platform. The SoC asserts the signal during power-up and when software initiates a hard reset sequence through the Reset Control (RST_CNT) register.

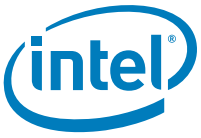
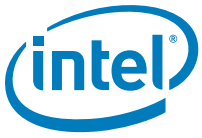


Table 90. PMC Signals (Sheet 2 of 2)

Signal Name	Direction /Type	Description
PMC_PWRBTN_N	I/O/ GPIOMV, MS	**Power Button:** The signal will cause SMI_N 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 the signal is pressed for more than 4 seconds, 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.
PMC_RSMRST_N	I/GPIOMV, MS	**Resume Well Reset:** Used for resetting the resume well. An external RC circuit is required to guarantee that the resume well power is valid prior to this signal going high.
PMC_RSTBTN_N	I/O/ GPIOMV, MS	**System Reset:** This signal forces an internal reset after being debounced. *This signal is muxed and may be used by other functions.*
PMC_SLP_S0IX_N	I/O/ GPIOMV, MS	**S0ix Sleep Control:** This signal is for power plane control. It can be used to control system power when it is in a S0ix state.
PMC_SLP_S4_N	I/O/ GPIOMV, MS	**S4 Sleep Control:** This signal is for power plane control. It can be used to control system power when it is in a S4 (Suspend to Disk) or S5 (Soft Off) state.
PMC_SUS_STAT_N	I/O/ GPIOMV, MS	**Suspend Status:** This signal is asserted by the SoC to indicate that the system will be entering a low power state soon. This can be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It can also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes. *This signal is muxed and may be used by other functions.*
PMC_SUSCLK	I/O/ GPIOMV, MS	**Suspend Clock:** This 32 kHz clock is an output of the RTC generator circuit for use by other chips for refresh clock. *This signal is muxed and may be used by other functions.*
PMC_SUSPWRDNACK	I/O/ GPIOMV, MS	**Suspend Power Down Acknowledge:** Asserted by the SoC when it does not require its Suspend well to be powered. This pin requires a pull-up to UNCORE_V1P8_G3. *This signal is muxed and may be used by other functions.*



19.3.2 Features

19.3.2.1 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 GEN_PMCON1.AG3E 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 two possible events that will wake the system after a power failure.

- **PMC_PWRBTN_N:** PMC_PWRBTN_N is always enabled as a wake event. When RSMRST_N is low (G3 state), the PM1_STS_EN.PWRBTN_STS bit is reset. When the SoC exits G3 after power returns (PMC_RSMRST_N goes high), the PMC_PWRBTN_N signal is already high (because the suspend plane goes high before PMC_RSMRST_N goes high) and the PM1_STS_EN.PWRBTN_STS bit is 0b.
- **RTC Alarm:** The PM1_STS_EN.RTC_EN bit is in the RTC well and is preserved after a power loss. Like PM1_STS_EN.PWRBTN_STS the PM1_STS_EN.RTC_STS bit is cleared when PMC_RSMRST_N goes low.

The SoC monitors both PMC_CORE_PWROK and PMC_RSMRST_N to detect for power failures. If PMC_CORE_PWROK goes low, the GEN_PMCON1.PWR_FLR bit is set. If PMC_RSMRST_N goes low, GEN_PMCON1.SUS_PWR_FLR is set.

Table 91. Transitions Due to Power Failure

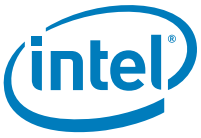
State at Power Failure	GEN_PMCON1.AG3E bit	Transition When Power Returns
S0	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0

19.3.2.2 Event Input Signals and Their Usage

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

19.3.2.2.1 PMC_PWRBTN_N (Power Button)

The PMC_PWRBTN_N signal operates as a "Fixed Power Button" as described in the Advanced Configuration and Power Interface specification. The signal has a 16 ms debounce on the input. The state transition descriptions are included in Table 92. Note that the transitions start as soon as the PMC_PWRBTN_N is pressed (but after the debounce logic), and does not depend on when the power button is released.



Note: During the time that the PMC_SLP_S4_N signal is stretched for the minimum assertion width (if enabled), the power button is not a wake event. Refer to note below for more details.

Table 92. Transitions Due to Power Button

Present State	Event	Transition / Action	Comment
S0/Cx	PMC_PWRBTN_N goes low	SMI_N or SCI generated (depending on PM1_CNT.SCI_EN, PM1_STS_EN.PWRBTN_EN and SMI_EN.GBL_SMI_EN)	Software typically initiates a Sleep state
S4/S5	PMC_PWRBTN_N goes low	Wake Event. Transitions to S0 state	Standard wakeup
G3	PMC_PWRBTN_N pressed	None	No effect since no power Not latched nor detected
S0, S4	PMC_PWRBTN_N held low for at least 4 consecutive seconds	Unconditional transition to S5 state	No dependence on processor or any other subsystem
S0ix	PMC_PWRBTN_N goes low	Wake Event. Transitions to S0 state	PM1_STS_EN.PWRBTN_EN should be set since a SMI/SCI event is required.

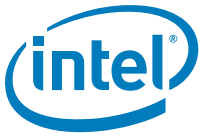
Power Button Override Function

If PMC_PWRBTN_N is observed active for at least four consecutive seconds, the state machine should unconditionally transition to the S5 state, regardless of present state (S0–S4), even if the PMC_CORE_PWROK is not active. In this case, the transition to the G2/S5 state should not depend on any particular response from the processor nor any similar dependency from any other subsystem.

The PMC_ PWRBTN_N status is readable to check if the button is currently being pressed or has been released. The status is taken after the de-bounce, and is readable using the GEN_PMCON2.PWRBTN_LVL bit.

Note: The 4-second PMC_PWRBTN_N assertion should only be used if a system lock-up has occurred. The 4-second timer starts counting when the SoC is in a S0 state. If the PMC_PWRBTN_N signal is asserted and held active when the system is in a suspend state (S4), the assertion causes a wake event. Once the system has resumed to the S0 state, the 4-second timer starts.

Note: During the time that the SLP_S4_N signal is stretched for the minimum assertion width (if enabled by GEN_PMCON1.S4ASE), the power button is not a wake event. As a result, it is conceivable that the user will press and continue to hold the power button waiting for the system to awake. Since a 4-second press of the power button is already defined as an unconditional power down, the power button timer will be forced to inactive while the power-cycle timer is in progress. Once the power-cycle timer has



expired, the power button awakes the system. Once the minimum PMC_SLP_S4_N power cycle expires, the power button must be pressed for another 4 to 5 seconds to create the override condition to S5.

19.3.2.2.2 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 SoC 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. See the Advanced Configuration and Power Interface specification for implementation details.

19.3.2.2.3 PME_B0 (PCI Power Management Event Bus 0)

The GPE0a_STS.PME_B0_STS bit exists to implement PME_N-like functionality for any internal device on Bus 0 with PCI power management capabilities.

19.3.2.2.4 PMC_RSTBTN_N Signal

When the PMC_RSTBTN_N pin is detected as active after the 16 ms debounce logic, the SoC attempts to perform a "graceful" reset, by waiting for the relevant internal devices to signal their idleness. If all devices are idle when the pin is detected active, the reset occurs immediately; otherwise, a counter starts. If at any point during the count all devices go idle the reset occurs. If the counter expires and any device is still active, a reset is forced upon the system even though activity is still occurring.

Once the reset is asserted, it remains asserted for 5 to 6 ms regardless of whether the PMC_RSTBTN_N input remains asserted or not. It cannot occur again until PMC_RSTBTN_N has been detected inactive after the debounce logic, and the system is back to a full S0 state with PMC_PLTRST_N inactive. Note that if RST_CNT.FULL_RST is set then PMC_RSTBTN_N will result in a full power cycle reset.

19.3.2.3 System Power Planes

The system has several independent power planes, as described in Table 93. Note that when a particular power plane is shut off, it should go to a 0 V level.

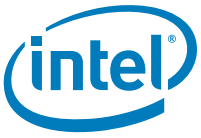


Table 93. System Power Planes

Plane	Controlled By	Description
Devices and Memory	PMC_SLP_S4_N	When PMC_SLP_S4_N goes active, power can be shut off to any circuit not required to wake the system from the S4/S5. Since the memory context does not need to be preserved in the S4/S5 state, the power to the memory can also be shut down. S4 and S5 requests are treated the same so no PMC_SLP_S5_N signal is implemented.
Devices	Implementation Specific	Individual subsystems may have their own power plane. For example, GPIO signals may be used to control the power to disk drives, audio amplifiers, or the display screen.
Suspend	PMC_SUSPWRDNACK	The suspend power planes are generally left on whenever the system has a charged main battery or is plugged in to AC power. In some cases, it may be preferable to disable the suspend power planes in S4/S5 states to save additional power. This requires some external logic (such as an embedded controller) to ensure that a wake event is still possible (such as the power button). When the SeC is enabled it is advised that the suspend power planes not be removed. Doing so may result in extremely long Sx exit times since the SeC if forced to consider it a cold boot which may, in turn, cause exit latency violations for software using the TXE.

19.3.2.3.1 Power Plane Control with PMC_SLP_S0IX_N and PMC_SLP_S4_N

The PMC_SLP_S0IX_N output signal can be used to cut power to any systems supplies that are not required during a S0ix system state.

Cutting power to the core may be done using the power supply, or by external FETs on the motherboard.

The PMC_SLP_S4_N 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.

19.3.2.3.2 PMC_SLP_S4_N and Suspend-To-RAM Sequencing

The system memory suspend voltage regulator is controlled by the Glue logic. The PMC_SLP_S4_N signal should be used to remove power to system memory. The PMC_SLP_S4_N logic in the SoC 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 GEN_PMCON1.S4ASE bit, the DRAM power must be controlled by the PMC_SLP_S4_N signal.

19.3.2.3.3 PMC_CORE_PWROK Signal

When asserted, PMC_CORE_PWROK is an indication to the SoC that its core well power rails are powered and stable. PMC_CORE_PWROK can be driven asynchronously. When PMC_CORE_PWROK is low, the SoC asynchronously asserts PMC_PLTRST_N. PMC_CORE_PWROK must not glitch, even if PMC_RSMRST_N is low.

It is required that the power rails associated with PCI Express have been valid for 99 ms prior to PWROK assertion in order to comply with the 100 ms T_{PVPERL} PCI Express 2.0 specification on PMC_PLTRST_N deassertion.

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

19.3.2.3.4 PMC_BATLOW_N (Battery Low)

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

19.3.2.4 SMI_N/SCI Generation

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

Once the SMI message has been delivered, the SoC 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 SoC will send another SMI 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 IRQs[11:9] or IRQs[23:20]. The interrupt polarity changes depending on whether it is on an interrupt shareable with a PIRQ or not. The interrupt remains asserted until all SCI sources are removed.

Table 94 shows which events can cause an SMI and SCI. Note that 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 94. Causes of SMI and SCI (Sheet 1 of 2)

Event	Status Indication[1]	Enable Condition	Interrupt Result			
			SMI_EN. GBL_SMI_EN=1b		SMI_EN. GBL_SMI_EN=0b	
			PM1_CNT .SCI_EN= 1b	PM1_CNT .SCI_EN= 0b	PM1_CNT. SCI_EN= 1b	PM1_CNT. SCI_EN=0 b
Power Button Override[3]	PM1_STS_EN. PWRBTNOR_STS	None	SCI	None	SCI	None
RTC Alarm	PM1_STS_EN. RTC_STS	PM1_STS_EN_EN. RTC_EN=1b	SCI	SMI	SCI	None
Power Button Press	PM1_STS_EN. PWRBTN_STS	PM1_STS_EN_EN. PWRBTN_EN=1b	SCI	SMI	SCI	None
SMI_EN.BIOS_RLS bit written to 1b[4]	PM1_STS_EN. GBL_STS	PM1_STS_EN_EN. GBL_EN=1b	SCI			
ACPI Timer overflow (2.34 seconds)	PM1_STS_EN. TMROF_STS	PM1_STS_EN_EN. TMROF_EN =1b	SCI	SMI	SCI	None
GPI[n][9]	GPE0a_STS. CORE_GPIO_STS[n][2] or GPE0a_STS. SUS_GPIO_STS[n][2]	GPIO_ROUT[n] = 10b & GPE0a_EN. CORE_GPIO_EN[n][2]=1b or GPE0a_EN. SUS_GPIO_EN[n][2]=1b	SCI	None	SCI	None
Internal, Bus 0, PME-Capable Agents (PME_B0)	GPE0a_STS. PME_B0_STS	GPE0_EN. PME_B0_EN=1b	SCI	SMI	SCI	None
BATLOW_N pin goes low	GPE0a_STS. BATLOW_STS_N	GPE0_EN. BATLOW_EN=1b	SCI	SMI	SCI	None
Software Generated GPE	GPE0a_STS. SWGPE_STS	GPE0_EN. SWGPE_EN=1b	SCI	SMI	SCI	None
DOSCI message from GUNIT[5]	GPE0a_STS. GUNIT_STS	None (enabled by G-Unit[8])	SCI	None	SCI	None
ASSERT_SMI message from SPI[5]	SMI_STS. SPI_SMI_STS	None (enabled by SPI controller)	SMI		None	
ASSERT_IS_SMI message from USB	SMI_STS. USB_IS_STS	SMI_EN. USB_IS_SMI_EN =1b	SMI		None	

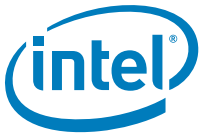


Table 94. Causes of SMI and SCI (Sheet 2 of 2)

Event	Status Indication[1]	Enable Condition	Interrupt Result			
			SMI_EN. GBL_SMI_EN=1b		SMI_EN. GBL_SMI_EN=0b	
			PM1_CNT .SCI_EN= 1b	PM1_CNT .SCI_EN= 0b	PM1_CNT. SCI_EN= 1b	PM1_CNT. SCI_EN=0 b
ASSERT_SMI message from USB	SMI_STS.USB_ST S	SMI_EN. USB_SMI_EN=1b	SMI		None	
ASSERT_SMI message from iLB[5]	SMI_STS. ILB_SMI_STS	None (enabled by iLB)	SMI		None	
Periodic timer expires	SMI_STS. PERIODIC_STS	SMI_EN. PERIODIC_EN=1b	SMI		None	
WDT first expiration	SMI_STS.TCO_ST S	SMI_EN.TCO_EN =1b	SMI		None	
64 ms timer expires	SMI_STS. SWSMI_TMR_STS	SMI_EN. SWSMI_TMR_EN =1b	SMI		None	
PM1_CNT.SLP_EN bit written to 1b	SMI_STS. SMI_ON_SLP_EN_ STS	SMI_EN. SMI_ON_SLP_EN =1b	Sync SMI[6]		None	
PM1_CNT.GBL_RL S written to 1b	SMI_STS.BIOS_ST S	SMI_EN. BIOS_EN=1b	Sync SMI[6]		None	
DOSMI message from GUNIT[5]	SMI_STS. GUNIT_SMI_STS	None (enabled by G-Unit[8])	SMI		None	
ASSERT_IS_SMI message from iLB[5]	SMI_STS. ILB_SMI_STS	None (enabled by iLB)	Sync SMI[7]		None	
GPI[n][10]	ALT_GPIO_SMI. CORE_GPIO_SMI_ STS[n][2] or ALT_GPIO_SMI. SUS_GPIO_SMI_S TS[n][2]	GPIO_ROUT[n]=0 1b & ALT_GPIO_SMI. CORE_GPIO_SMI _EN[n][2]=1b or ALT_GPIO_SMI. SUS_GPIO_SMI_E N[n][2]=1b	SMI		None	
USB Per-Port Registers Write Enable bit is changed from 0b to 1b	UPRWC.WE_STS & SMI_STS. USB_IS_STS	UPRWC. WE_SMI_E=1b & SMI_EN. USB_IS_SMI_EN =1b	Sync SMI[6]		None	

NOTES:
1. Most of the status bits (except otherwise is noted) are set according to event occurrence regardless to the enable bit.

2. GPIO status bits are set only if enable criteria is true. GPIO_ROUT[n]=10b & GPE0a_EN.x_GPIO_EN[n] for GPE0a_STS.x_GPIO_STS[n] (SCI). GPIO_ROUT[n]=01b & ALT_GPIO_SMI. x_GPIO_SMI_EN[n]=1b for ALT_GPIO_SMI.x_GPIO_SMI_STS[n] (SMI).
3. When power button override occurs, the system will transition immediately to S5. The SCI will only occur after the next wake to S0 if the residual status bit (PM1_STS_EN.PWRBTNOR_STS) is not cleared prior to setting PM1_CNT.SCI_EN.
4. PM1_STS_EN.GBL_STS being set will cause an SCI, even if the PM1_CNT.SCI_EN bit is not set. Software must take great care not to set the SMI_ENBIOS_RLS bit (which causes PM1_STS_EN.GBL_STS to be set) if the SCI handler is not in place.
5. No enable bits for these SCI/SMI messages in the PMC. Enable capability should be implemented in the source unit.
6. Sync SMI has the same message opcode toward T-Unit. Special treatment regarding this Sync SMI is holding completion to host till SYNC_SMI_ACK message is received from T-Unit.
7. Sync SMI has the same message opcode toward T-Unit. Special treatment regarding this Sync SMI is holding the SSMI_ACK message to iLB till SYNC_SMI_ACK message is received from T-Unit.
8. The G-Unit is an internal functional sub-block which forms part of the graphics functional block.
9. The GPE0a_STS.CORE_GPIO_STS[31:24] & GPE0a_EN.CORE_GPIO_EN[31:24] register bits correspond to GPIO_S0_SC[7:0]. GPE0a_STS.SUS_GPIO_STS[23:16] & GPE0a_EN.SUS_GPIO_EN[23:16] correspond to GPIO_S5[7:0].
10. The ALT_GPIO_SMI.CORE_GPIO_SMI_STS[31:24] & ALT_GPIO_SMI.CORE_GPIO_SMI_EN[15:8] register bits correspond to GPIO_S0_SC[7:0]. ALT_GPIO_SMI.SUS_GPIO_SMI_STS[23:16] & ALT_GPIO_SMI.SUS_GPIO_SMI_EN[7:0] correspond to GPIO_S5[7:0].

19.3.2.5 Platform Clock Support

The SoC supports up to 6 clocks (PMC_PLT_CLK[5:0])with a frequency of 19.2 MHz. These clocks are available for general system use, where appropriate and each have Control & Frequency register fields associated with them.

19.3.2.6 INIT_N (Initialization) Generation

The INIT_N functionality is implemented as a ‘virtual wire’ internal to the SoC rather than a discrete signal. This virtual wire is asserted based on any one of the events described in below table. When any of these events occur, INIT_N is asserted for 16 PCI clocks and then driven high.

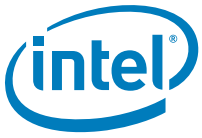
INIT_N, when asserted, resets integer registers inside the CPU cores without affecting its internal caches or floating-point registers. The cores then begin execution at the power on Reset vector configured during power on configuration.

Table 95. INIT_N Assertion Causes

Cause
PORT92.INIT_NOW transitions from 0b to1b.
RST_CNT.SYS_RST = 0b and RST_CNT.RST_CPU transitions from 0b to 1b

19.3.3 References

Advanced Configuration and Power Interface Specification, Revision 3.0: http://www.acpi.info/



19.4 PCU - Fast Serial Peripheral Interface (SPI)

The SoC implements a SPI controller as the interface for BIOS Flash storage. This SPI Flash device is also required to support configuration storage for the firmware for the Trusted Execution Engine. The controller supports a maximum of two SPI Flash devices, using two chip select signals, with speeds of 14.28MHz, 20MHz, 25MHz, 40MHz or 50MHz and both have to be Fast SPI.

SoC Supports FAST SPI mode.

Note: The default interface speed is 20 MHz.

SPI 'Fast mode' is quad mode.

19.4.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

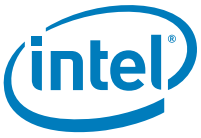
The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 96. SPI Signals

Signal Name	Direction / Type	Description
FST_SPI_CLK	I/O GPIO	**Fast SPI Clock:** When the bus is idle, the owner will drive the clock signal low.
FST_SPI_CS[0]_N	I/O GPIO	**Fast SPI Chip Select 0:** Used as the SPI bus request signal for the first SPI Flash device.
FST_SPI_CS[1]_N	I/O GPIO	**Fast SPI Chip Select 1:** Used as the SPI bus request signal for the second SPI Flash devices.
FST_SPI_CS[2]_N	I/O GPIO	**Fast SPI Chip Select 2:** Used as the SPI bus request signal for the second SPI Flash devices.
FST_SPI_D[3:0]	I/O GPIO	**Fast SPI Data Pad:** Data Input/output pin for the SoC.

Note: All SPI signals are tri-stated until PMC_CORE_PWROK is asserted.



19.4.2 Features

1)Descriptor Mode Capabilities

a)Two modes of operation

i)Descriptor mode with security access restrictions

ii)Non-Descriptor mode, no access security restrictions (ICH7 style)

(1)BIOS Only

(2)If the SPI Flash Signature is invalid, the SPI flash operates in non-descriptor mode

b)Supports Flash that is divided into 5 regions and accessible by 3 masters

i)Regions (5)

(1) Flash Descriptor and Chipset Soft Straps

(2) BIOS

(3) TXE

(4) Platform Data

ii)Masters (3)

(1) Host CPU (for BIOS)

(2) TXE

iii)Regions are allowed to extend across multiple Flash components

iv)Regions are aligned to 4K blocks/sectors

b)Chipset Soft Strap region provides the ability to use Flash NVM as an alternative to hardware pullup/pulldown resistors for both SoC and the processor Complex

i)Each Unit that pulls Soft straps from SPI should have a default value that is used if the Flash Signature is invalid.

b)The top of the Flash Descriptor contains the Flash Upper Map

ii)This is used by software to define Flash vendor specific capabilities

b)The top 256B of the flash descriptor is reserved for use by the OEM

2)Security Capabilities

a)Descriptor based Region Restriction: Hardware enforced security restricting master accesses to different regions

i)Flash Descriptor region settings define separate read/write access to each region per master.

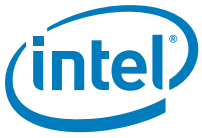
ii)Uses SAI for master accesses security checking

(1)Soft Strap+fuse to disable sourceID and SAI checks

iii)Flash Security Override Pin Strap

(1)Removes all descriptor based security

(2)Disables the write protection to the BIOS Protected Range 4 (PR4).



iv)Each master can grant other masters read/write access to its region

b)Protected Range Registers.

i)3 sets (one for each master) of Lockable Protected Range registers that can restrict program register accesses from the same master.

ii)Can span multiple regions

iii)Separate read and write protection

iv)Special case: BIOS PR4 write protect values are received from Soft Strap and affect all masters.

c)SMI Write Protection for BIOS

i)If enabled, will cause an SMI if a program register access occurs. The primary purpose of this requirement is to support SMI based BIOS update utilities.

d)Illegal Instruction protection for instructions such as Chip Erase

e)Lockable software sequencing opcodes

3)SPI Flash Access

a)Direct Read Access

b)Program Register Access

i)Hardware Sequencing

(1)Software Sequencing uses HW to provide the basic instructions of read, write, and erase.

ii)Software Sequencing

(1)Allows SW to use any legal Opcode

c)Support for Boot BIOS on SPI.

i)Non-boot BIOS that is accessible through program register only can be used on SPI when boot BIOS is located on some other interface.

d)Pre-fetching/Caching to improve performance

i)Separate 64B pre-fetch/cache each for HOST and SEC direct read accesses

4)SFDP Parameter Discoverability[1]

5)Flash Component Capabilities

a)In Descriptor mode, supports two SPI Flash components using two separate chip select pins, CS0# and CS1#. Only one component supported in non-descriptor mode.

i)Components must have the same erasable block/sector size

ii)Each component can be up to 16MB (32MB total addressable) using 24-bit addressing.

b)1.8V SPI I/O buffer VCC

c)Supports the SPI Fast Read/Write instruction and frequencies of 20MHz, 33MHz and 50 MHz. Supports the SPI Dual Output Fast Read/Write instruction with frequencies of 20 MHz, 33 MHz and 50 MHz

d)Supports the SPI Quad Output Fast Read/Write instruction with frequencies of 20 MHz, 33 MHz and 50 MHz

e)Uses standardized Flash Instruction Set.

f)Supports non-power of 2 flash sizes, with the following restrictions:

i)Only supported in Descriptor Mode.

ii)BIOS accesses in non-descriptor mode to a non-binary flash size will not function properly.

iii)The Flash Regions must be programmed to the actual size of the Flash Component(s).

iv)If using two flash components, the 1st flash component (the one with the Flash Descriptor) must be of binary size. The 2nd flash component can be a non-binary size. If using only one flash component, it can be of non-binary size.

v)The value programmed in the Flash Descriptor Component Density must be set to the next power of 2 value larger than the non-binary size.

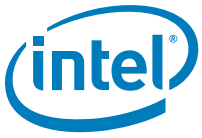
8)Reset Capabilities

a)RSMRST#

i)When RSMRST# is asserted, SoC will tri-state with a weak pull-up all SPI pins

i)The SPI Controller will implement a sideband handshake((handshake is reset warn message)) with PMC when a host reset is requested to allow the SPI Flash controller to complete any outstanding atomic sequences and quiescence the SPI Bus

Note: There is no N*parameter headers support on SoC, DTR and 32-bit addressing is not supported



19.5 PCU - Universal Asynchronous Receiver / Transmitter (UART)

This section describes the Universal Asynchronous Receiver/Transmitter (UART) serial port integrated into the PCU. The UART may be controlled through programmed IO.

Note: Only a minimal ball-count, comprising receive & transmit signals, UART port is implemented. Further, a maximum baud rate of only 115,200 bps is supported. For this reason, it is recommended that the UART port be used for debug purposes only.

19.5.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 97. UART Signals

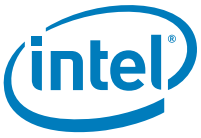
Signal Name	Direction / Type	Description
UART0_DATAIN	I/GPIOHV, HS	**COM1 Receive:** Serial data input from device pin to the receive port. *This signal is muxed and may be used by other functions.*
UART0_DATAOUT	O/GPIOHV, HS	**COM1 Transmit:** Serial data output from transmit port to the device pin. *This signal is muxed and may be used by other functions.*

19.5.2 Features

The serial port consists of a UART which supports a subset of the functions of the 16550 industry standard.

The UART performs serial-to-parallel conversion on data characters received from a peripheral device and parallel-to-serial conversion on data characters received from the processor. The processor may read the complete status of the UART at any time during the functional operation. Available status information includes the type and condition of the transfer operations being performed by the UART and any error conditions.

The serial port may operate in either FIFO or non-FIFO mode. In FIFO mode, a 16-byte transmit FIFO holds data from the processor to be transmitted on the serial link and a 16-byte Receive FIFO buffers data from the serial link until read by the processor.



The UART includes a programmable baud rate generator which is capable of generating a baud rate of between 50 bps and 115,200 bps from a fixed baud clock input of 1.8432 MHz. The baud rate is calculated as follows:

Baud Rate Calculation:

$$\text{BaudRate} = \frac{1.8432 \times 10^{6}}{16 \times \text{Divisor}}$$

The divisor is defined by the Divisor Latch LSB and Divisor Latch MSB registers. Some common values are shown in Table 98.

Table 98. Baud Rate Examples

Desired Baud Rate	Divisor	Divisor Latch LSB Register	Divisor Latch MSB Register
115,200	1	1h	0h
57,600	2	2h	0h
38,400	3	3h	0h
19,200	6	6h	0h
9,600	12	Ch	0h
4,800	24	18h	0h
2,400	48	30h	0h
1,200	96	60h	0h
300	384	80h	1h
50	2,304	0h	9h

The UART has interrupt support and those interrupts may be programmed to the user's requirements, minimizing the computing required to handle the communications link. Each UART may operate in a polled or an interrupt driven environment as configured by software.

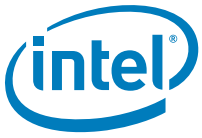
19.5.2.1 FIFO Operation

19.5.2.1.1 FIFO Interrupt Mode Operation

Receiver Interrupt

When the Receive FIFO and receiver interrupts are enabled (FIFO Control Register, bit 0 = 1b and Interrupt Enable Register (IIR), bit 0 = 1b), receiver interrupts occur as follows:

- The receive data available interrupt is invoked when the FIFO has reached its programmed trigger level. The interrupt is cleared when the FIFO drops below the programmed trigger level.



- The IIR receive data available indication also occurs when the FIFO trigger level is reached, and like the interrupt, the bits are cleared when the FIFO drops below the trigger level.
- The receiver line status interrupt (IIR = C6h), as before, has the highest priority. The receiver data available interrupt (IIR = C4h) is lower. The line status interrupt occurs only when the character at the top of the FIFO has errors.
- The COM1_LSR.DR bit is set to 1b as soon as a character is transferred from the shift register to the Receive FIFO. This bit is reset to 0b when the FIFO is empty.

Character Time Out Interrupt

When the receiver FIFO and receiver time out interrupt are enabled, a character time out interrupt occurs when all of the following conditions exist:

- At least one character is in the FIFO.
- The last received character was longer than four continuous character times ago (if 2 stop bits are programmed the second one is included in this time delay).
- The most recent processor read of the FIFO was longer than four continuous character times ago.
- The receiver FIFO trigger level is greater than one.

The maximum time between a received character and a timeout interrupt is 160 ms at 300 baud with a 12-bit receive character (i.e., 1 start, 8 data, 1 parity, and 2 stop bits).

When a time out interrupt occurs, it is cleared and the timer is reset when the processor reads one character from the receiver FIFO. If a time out interrupt has not occurred, the time out timer is reset after a new character is received or after the processor reads the receiver FIFO.

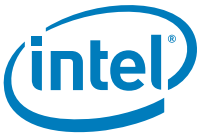
Transmit Interrupt

When the transmitter FIFO and transmitter interrupt are enabled (FIFO Control Register, bit 0 = 1b and Interrupt Enable Register, bit 0 = 1b), transmit interrupts occur as follows:

The Transmit Data Request interrupt occurs when the transmit FIFO is half empty or more than half empty. The interrupt is cleared as soon as the Transmit Holding Register is written (1 to 16 characters may be written to the transmit FIFO while servicing the interrupt) or the Interrupt Identification Register is read.

19.5.2.1.2 FIFO Polled Mode Operation

With the FIFOs enabled (FIFO Control register, bit 0 = 1b), setting Interrupt Enable register (IER), bits 3:0 = 000b puts the serial port in the FIFO polled mode of operation. Since the receiver and the transmitter are controlled separately, either one or both may be in the polled mode of operation. In this mode, software checks receiver and transmitter status through the Line Status Register (LSR). As stated in the register description:



- LSR[0] is set as long as there is one byte in the receiver FIFO.
- LSR[1] through LSR[4] specify which error(s) has occurred for the character at the top of the FIFO. Character error status is handled the same way as interrupt mode. The Interrupt Identification Register is not affected since IER[2] = 0b.
- LSR[5] indicates when the transmitter FIFO needs data.
- LSR[6] indicates that both the transmitter FIFO and shift register are empty.
- LSR[7] indicates whether there are any errors in the receiver FIFO.

19.5.3 Use

19.5.3.1 Base I/O Address

19.5.3.1.1 COM1

The base I/O address for the COM1 UART is fixed to 3F8h.

19.5.3.2 Legacy Interrupt

19.5.3.2.1 COM1

The legacy interrupt assigned to the COM1 UART is fixed to IRQ4.

19.5.4 UART Enable/Disable

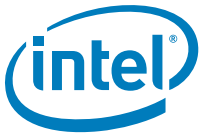
The COM1 UART may be enabled or disabled using the UART_CONT.COM1EN register bit. By default, the UART is disabled.

Note: It is recommended that the UART be disabled during normal platform operation. An enabled UART can interfere with platform power management.

19.5.5 IO Mapped Registers

There are 12 registers associated with the UART. These registers share eight address locations in the IO address space. Table 99 shows the registers and their addresses as offsets of a base address. Note that the state of the COM1_LCR.DLAB register bit, which is the most significant bit (MSB) of the Serial Line Control register, affects the selection of certain of the UART registers. The COM1_LCR.DLAB register bit must be set high by the system software to access the Baud Rate Generator Divisor Latches.

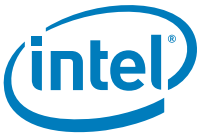
Notes:
1. These registers are consolidated in the Receiver Buffer / Transmitter Holding Register (COM1_RX_TX_BUFFER)
2. These registers are consolidated in the Interrupt Enable Register (COM1_IER)
3. These registers are consolidated in the Interrupt Identification / FIFO Control Register (COM1_IIR)
4. These registers are implemented but unused since the UART signals related to modem interaction are not implemented.



19.6 Register Map

Table 99. Register Access List

Register Address (Offset to Base IO Address)	COM1_LCR.DLAB Value	Register Access Type	Register Accessed
0h	0b	RO	Receiver Buffer[1]
0h	0b	WO	Transmitter Holding[1]
0h	1b	RW	Divisor Latch LSB (Lowest Significant Bit)[1]
1h	0b	RW	Interrupt Enable[2]
1h	1b	RW	Divisor Latch MSB (Most Significant Bit)[2]
2h	xb	RO	Interrupt Identification[3]
2h	xb	WO	FIFO Control[3]
3h	xb	RW	Line Control
4h	xb	RW	Modem Control[4]
5h	xb	RO	Line Status
6h	xb	RO	Modem Status[4]
7h	xb	RW	Scratchpad



19.7 PCU - Intel Legacy Block (iLB) Overview

The Intel Legacy Block (iLB) is a collection of disparate functional blocks that are critical for implementing the legacy PC platform features. These blocks include:

- "PCU - iLB - Low Pin Count (LPC) Bridge"
- "PCU - iLB - Real Time Clock (RTC)"
- "PCU - iLB - 8254 Timers"
- "PCU - iLB - High Precision Event Timer (HPET)"
- "PCU - iLB - GPIO"
- "PCU - iLB - IO APIC"
- "PCU - iLB - 8259 Programmable Interrupt Controllers (PIC)"

The iLB also implements a register range for configuration of some of those blocks along with support for Non-Maskable Interrupts (NMI).

19.7.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details as well as the subsequent sections.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 100. iLB Signals

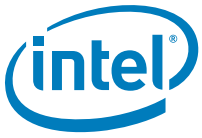
Signal Name	Direction / Type	Description
NMI_N	I/GPIOMV, MS	Non-Maskable Interrupt: This is an NMI event indication into the SoC. *This signal is muxed and may be used by other functions.*

19.7.2 Features

19.7.2.1 Key Features

The key features of various blocks are as follows:

- LPC Interface
  - — Supports Low Pin Count (LPC) 1.1 Specification
  - — No support for DMA or bus mastering
  - — Supports Trusted Platform Module (TPM) 1.2



- General Purpose Input Output
  - Legacy control interface for SoC GPIOs
  - I/O mapped registers
- 8259 Programmable Interrupt Controller
  - Legacy interrupt support
  - 15 total interrupts through two cascaded controllers
  - I/O mapped registers
- I/O Advanced Programmable Interrupt Controller
  - Legacy-free interrupt support
  - 115 total interrupts
  - Memory mapped registers
- 8254
  - Legacy timer support
  - Three timers with fixed uses: System Timer, Refresh Request Signal and Speaker Tone
  - I/O mapped registers
- HPET - High Performance Event Timers
  - Legacy-free timer support
  - Three timers and one counter
  - Memory mapped registers
- Real-Time Clock (RTC)
  - 242 byte RAM backed by battery (aka CMOS RAM)
  - Can generate wake/interrupt when time matches programmed value
  - I/O and indexed registers

19.7.2.2 Non-Maskable Interrupt

NMI support is enabled by setting the NMI Enable (NMI_EN) bit, at IO Port 70h, Bit 7, to 1b.

Non-Maskable Interrupts (NMIs) can be generated by several sources, as described in Table 101.

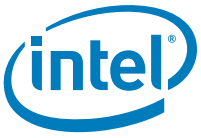


Table 101. NMI Sources

NMI Source	NMI Source Enabler / Disabler	NMI Source Status	Alternate Configuration
SERR# goes active **NOTE:** A SERR# is only generated internally in the SoC)	NSC.SNE	NSC.SNS	All NMI sources may, alternatively, generate a SMI by setting GNMI.NMI2SMIEN=1b The SoC uses GNMI.NMI2SMIST for observing SMI status
IOCHK# goes active **NOTE:** A IOCHK# is only generated as a SERIRQ# frame	NSC.INE	NSC.INS	
NMI goes active **NOTE:** Active can be defined as being on the positive or negative edge of the signal using the GNMI.GNMIED register bit.	GNMI.GNMIED	GNMI.GNMIS	
Software sets the GNMI.NMIN register bit	GNMI.NMIN	GNMI.NMINS	

19.7.2.3 S0ix Support

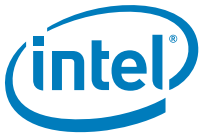
There is no requirement to set “HPET_GCFG.EN” to 0b. Basically turn off HPET during S0i2/3. RTD3hot status is not a key requirement for OS anymore.

The S1 state described in the HPET spec is a "CPU Stop Grant" condition. This condition is met during the S0i2/3 states, (although entry into S0i2/3 is performed in a different way).

19.7.3 Use

19.7.3.1 S0ix Support

Prior to entry into S0i2 or S0i3 state, the driver/OS must set HPET_GCFG.EN to 0b to indicate RTD3hot status.



19.8 PCU - iLB - Low Pin Count (LPC) Bridge

The SoC implements an LPC Interface as described in the LPC 1.1 Specification. The Low Pin Count (LPC) bridge function of the SoC resides in PCI Device 31, Function 0.

Note: In addition to the LPC bridge interface function, D31:F0 contains other functional units including interrupt controllers, timers, power management, system management, GPIO, and RTC.

19.8.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 102. LPC Signals

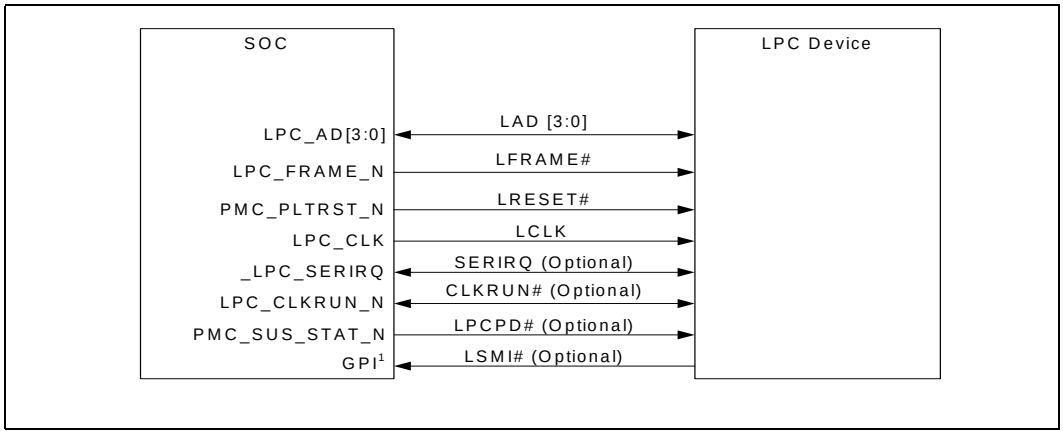
Signal Name	Direction / Type	Description
LPC_AD[3:0]	I/O/ GPIOHV, HS	**LPC Multiplexed Command, Address, Data:** Internal pull-ups are provided for these signals. *These signals are muxed and may be used by other functions.*
LPC_CLKOUT[0]	I/O/ GPIOHV, HS	**LPC Clock [0] Out:** 19MHz PCI-like clock driven to LPC peripherals. *These signals are muxed and may be used by other functions.*
LPC_CLKOUT[1]	I/O/ GPIOHV, HS	**LPC Clock [1] Out:** 19MHz PCI-like clock driven to LPC peripherals. Can be configured as an input to compensate for board routing delays through Soft Strap. *These signals are muxed and may be used by other functions.*
LPC_CLKRUN_N	I/O/ GPIOHV, HS	**LPC Clock Run:** Input to determine the status of LPC_CLK and an open drain output used to request starting or speeding up LPC_CLK. This is a sustained tri-state signal used by the central resource to request permission to stop or slow LPC_CLK. The central resource is responsible for maintaining the signal in the asserted state when LPC_CLK is running and deasserts the signal to request permission to stop or slow LPC_CLK. An internal pull-up is provided for this signal. *This signal is muxed and may be used by other functions.*
LPC_FRAME_N	I/O/ GPIOHV, HS	**LPC Frame:** This signal indicates the start of an LPC cycle, or an abort. *This signal is muxed and may be used by other functions.*
LPC_SERIRQ	I/O/ GPIOHV, HS	**Serial Interrupt Request:** This signal implements the serial interrupt protocol. *This signal is muxed and may be used by other functions.* *Note: A level shifter needs to be implemented on this signal.*

19.8.2 Features

The LPC interface to the SoC is shown in Figure 32. Note that the SoC implements all of the signals that are shown as optional, but peripherals are not required to do so.

Note: The LPC controller does not implement bus mastering cycles or DMA.

Figure 32. LPC Interface Diagram



NOTES:
1. The General Purpose Input (GPI) must use a SMI capable GPIO: GPIO_S0_SC[7:0].

19.8.2.1 Memory Cycle Notes

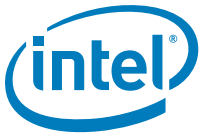
For cycles below 16M, the LPC Controller will perform standard LPC memory cycles. For cycles targeting firmware (BIOS/EFI code only), firmware memory cycles are used. Only 8-bit transfers are performed. If a larger transfer appears, the LPC controller will break it into multiple 8-bit transfers until the request is satisfied.

If the cycle is not claimed by any peripheral (and subsequently aborted), the LPC Controller will return a value of all 1's to the CPU.

19.8.2.2 Trusted Platform Module (TPM) 1.2 Support

The LPC interface supports accessing Trusted Platform Module (TPM) 1.2 devices via the LPC TPM START encoding. Memory addresses within the range FED00000h to FED40FFFh will be accepted by the LPC Bridge and sent on LPC as TPM special cycles. No additional checking of the memory cycle is performed.

Note: This is different to the FED00000h to FED4BFFFh range implemented on some other Intel components since no Intel® Trusted Execution Technology (Intel® TXT) transactions are supported.



19.8.2.3 FWH Cycle Notes

If the LPC controller receives any SYNC returned from the device other than short (0101), long wait (0110), or ready (0000) when running a FWH cycle, indeterminate results may occur. A FWH device is not allowed to assert an Error SYNC.

BIOS/EFI boot from LPC is not supported when Secure Boot is enabled.

19.8.2.4 Subtractive Decode

All cycles that are not decoded internally, and are not targeted for LPC (i.e., configuration cycles, IO cycles above 64KB and memory cycles above 16MB), will be sent to LPC with LPC_FRAME_N not asserted.

19.8.2.5 POST Code Redirection

Writes to addresses 80h - 8Fh in IO register space will also be passed to the LPC bus.

Note: Reads of these addresses do not result in any LPC transactions.

19.8.2.6 Power Management

19.8.2.6.1 LPCPD_N Protocol

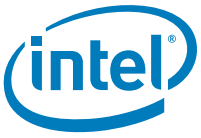
Same timings as for PMC_SUS_STAT_N. After driving PMC_SUS_STAT_N active, the SoC drives LPC_FRAME_N low, and tri-states (or drives low) LPC_AD[3:0].

Note: The Low Pin Count Interface Specification, Revision 1.1 defines the LPCPD_N protocol where there is at least 30 µs from LPCPD_N assertion to LRST_N assertion. This specification explicitly states that this protocol only applies to entry/exit of low power states which does not include asynchronous reset events. The SoC asserts both PMC_SUS_STAT_N (connects to LPCPD_N) and PLTRST_N (connects to LRST_N) at the same time during a global reset. This is not inconsistent with the LPC LPCPD_N protocol.

19.8.2.6.2 Clock Run (CLKRUN)

When there are no pending LPC cycles, and SERIRQ is in quiet mode, the SoC can shut down the LPC clock. The SoC indicates that the LPC clock is going to shut down by de-asserting the LPC_CLKRUN_N signal. LPC devices that require the clock to stay running should drive LPC_CLKRUN_N_N low within 4 clocks of its de-assertion. If no device drives the signal low within 4 clocks, the LPC clock will stop. If a device asserts LPC_CLKRUN_N, the SoC will start the LPC clock and assert LPC_CLKRUN_N.

Note: The CLKRUN protocol is disabled by default. See Section 19.8.3.2.2, “Clock Run Enable” on page 236 for further details.



19.8.2.7 Serialized IRQ (SERIRQ)

19.8.2.7.1 Overview

The interrupt controller supports a serial IRQ scheme. The signal used to transmit this information is shared between the interrupt controller and all peripherals that support serial interrupts. The signal line, LPC_SERIRQ, is synchronous to LPC clock, and follows the sustained tri-state protocol that is used by LPC signals. The serial IRQ protocol defines this sustained tri-state signaling in the following fashion:

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

The interrupt controller supports 21 serial interrupts. These represent the 15 ISA interrupts (IRQ0- 1, 3-15), the four PCI interrupts, and the control signals SMI_N and IOCHK_N. Serial interrupt information is transferred using three types of frames:

- Start Frame: LPC_SERIRQ line driven low by the interrupt controller to indicate the start of IRQ transmission
- Data Frames: IRQ information transmitted by peripherals. The interrupt controller supports 21 data frames.
- Stop Frame: LPC_SERIRQ line driven low by the interrupt controller to indicate end of transmission and next mode of operation.

19.8.2.7.2 Start Frame

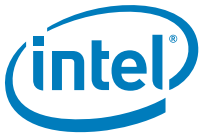
The serial IRQ protocol has two modes of operation which affect the start frame:

- Continuous Mode: The interrupt controller is solely responsible for generating the start frame
- Quiet Mode: Peripheral initiates the start frame, and the interrupt controller completes it.

These modes are entered via the length of the stop frame.

Continuous mode must be entered first, to start the first frame. This start frame width is 8 LPC clocks. This is a polling mode.

In Quiet mode, the LPC_SERIRQ line remains inactive and pulled up between the Stop and Start Frame until a peripheral drives LPC_SERIRQ low. The interrupt controller senses the line low and drives it low for the remainder of the Start Frame. Since the first LPC clock of the start frame was driven by the peripheral, the interrupt controller drives LPC_SERIRQ low for 1 LPC clock less than in continuous mode. This mode of operation allows for lower power operation.



19.8.2.7.3 Data Frames

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

- **Sample Phase:** During this phase, a device drives LPC_SERIRQ low if its corresponding interrupt signal is low. If its corresponding interrupt is high, then the LPC_SERIRQ devices tri-state LPC_SERIRQ. LPC_SERIRQ remains high due to pull-up resistors.
- **Recovery Phase:** During this phase, a device drives LPC_SERIRQ high if it was driven low during the Sample Phase. If it was not driven during the sample phase, it remains tri-stated in this phase.
- **Turn-around Phase:** The device tri-states LPC_SERIRQ.

19.8.2.7.4 Stop Frame

After the data frames, a Stop Frame will be driven by the interrupt controller. LPC_SERIRQ will be driven low for two or three LPC clocks. The number of clocks is determined by the SCNT.MD register bit. The number of clocks determines the next mode, as indicated in Table 103.

Table 103. SERIRQ, Stop Frame Width to Operation Mode Mapping

Stop Frame Width	Next Mode
Two LPC clocks	**Quiet Mode:** Any SERIRQ device initiates a Start Frame
Three LPC clocks	**Continuous Mode:** Only the interrupt controller initiates a Start Frame

19.8.2.7.5 Serial Interrupts Not Supported

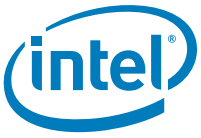
There are four interrupts on the serial stream which are not supported by the interrupt controller. These interrupts are:

- IRQ0: Heartbeat interrupt generated off of the internal 8254 counter 0.
- IRQ8: RTC interrupt can only be generated internally.
- IRQ13: This interrupt (floating point error) is not supported.

The interrupt controller will ignore the state of these interrupts in the stream.

19.8.2.7.6 Data Frame Format and Issues

Table below shows the format of the data frames. The decoded INT[A:D]_N values are ANDed with the corresponding PCI-express input signals (PIRQ[A:D]_N). This way, the interrupt can be shared.



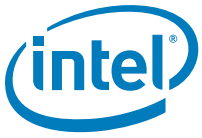
The other interrupts decoded via SERIRQ are also ANDed with the corresponding internal interrupts. For example, if IRQ10 is set to be used as the SCI, then it is ANDed with the decoded value for IRQ10 from the SERIRQ stream.

Table 104. SERIRQ Interrupt Mapping

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
1	IRQ0	2	Ignored. Can only be generated via the internal 8524
2	IRQ1	5	Before port 60h latch
3	SMI_N	8	Causes SMI_N if low. Sets SMI_STS.ILB_SMI_STS register bit.
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	
8	IRQ7	23	
9	IRQ8	26	Ignored. IRQ8_N can only be generated internally
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	Before port 60h latch
14	IRQ13	41	Ignored.
15	IRQ14	44	Ignored
16	IRQ15	47	
17	IOCHCK_N	50	Same as ISA IOCHCK_N going active.
18	PCI INTA_N	53	
19	PCI INTB_N	56	
20	PCI INTC_N	59	
21	PCI INTD_N	62	

19.8.2.7.7 S0ix Support

During S0i2 & S0i3, the LPC and SERIRQ interfaces are disabled.



19.8.3 Use

19.8.3.1 LPC Clock Delay Compensation

In order to meet LPC interface AC timing requirements, a LPC clock loop back is required. The operation of this loop back can be configured in two ways:

1. On the SOC: In this configuration, LPC_CLK[0] is looped back on itself on the SOC pad.

   a. Benefit:

   LPC_CLK[0] and LPC_CLK[1] are both available for system clocking

   b. Drawback:

   Clock delay compensation is less effective at compensating for mainboard delay

   c. Soft Strap & Register Requirements:

   Soft Strap LPCCLK_SLC = 0b

   Configuration is reflected by register bit LPCC.LPCCLK_SLC=0b

   Soft Strap LPCCLK1_ENB = 0b (LPC_CLK[1] disabled) or 1b (LPC_CLK[1] enabled)

2. Configuration is reflected by register bit LPCC.LPCCLK1EN=0b (LPC_CLK[1] disabled) or 1b (LPC_CLK[1] enabled)
3. On the main board: In this configuration, LPC_CLK[0] is looped back to LPC_CLK[1] on the main board.

   a. Benefit:

   Clock delay compensating in more effective at compensating for main board delay

   b. Drawback:

   Only LPC_CLK[0] is available for system clocking. LPC_CLK[1] must be disabled.

   c. Soft Strap & Register Requirements:

   Soft Strap LPCCLK_SLC = 1b

   Configuration is reflected by register bit LPCC.LPCCLK_SLC=1b

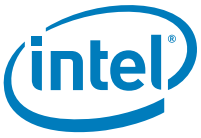
   Soft Strap LPCCLK1_ENB = 0b (LPC_CLK[1] disabled)

   Configuration is reflected by register bit LPCC.LPCCLK1EN=0b

19.8.3.2 LPC Power Management

19.8.3.2.1 Clock Enabling

The LPC clocks can be enabled or disabled by setting or clearing, respectively, the LPCC.LPCCLK[1:0]EN bits.



19.8.3.2.2 Clock Run Enable

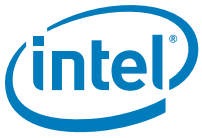
The Clock Run protocol is disabled by default and should only be enabled during operating system run-time, once all LPC devices have been initialized. The Clock Run protocol is enabled by setting the LPCC.CLKRUN_EN register bit.

19.8.3.3 SERIRQ Disable

Serialized IRQ support may be disabled by setting the OIC.SIRQEN bit to 0b.

19.8.4 References

- Low Pin Count Interface Specification, Revision 1.1 (LPC): http://www.intel.com/design/chipsets/industry/lpc.htm
- Serialized IRQ Support for PCI Systems, Revision 6.0: http://www.smsc.com/media/Downloads_Public/papers/serirq60.doc
- Implementing Industry Standard Architecture (ISA) with Intel® Express Chipsets (318244): http://www.intel.com/assets/pdf/whitepaper/318244.pdf



19.9 PCU - iLB - Real Time Clock (RTC)

The SoC contains a real-time clock with 242 bytes of battery-backed RAM. The real-time clock performs two key functions—keeping track of the time of day and storing system data, even when the system is powered down. The RTC operates on a 32.768 kHz crystal and a 3.3 V battery.

The RTC 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 supports a date alarm that allows for scheduling a wake up event up to 30 days in advance.

19.9.1 Signal Descriptions

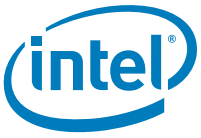
Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

Table 105. RTC Signals

Signal Name	Direction / Type	Description
RTC_X1	I Analog	**Crystal Input 1:** This signal is connected to the 32.768 kHz crystal. If no external crystal is used, the signal can be driven with the desired clock rate.
RTC_X2	I Analog	**Crystal Input 2:** This signal is connected to the 32.768 kHz crystal. If no external crystal is used, the signal should be left floating.
RTC_RST_N	I	**RTC Reset:** When asserted, this signal resets register bits in the RTC well. **NOTE:** Unless CMOS is being cleared (only to be done in the G3 power state), the signal input must always be high when all other RTC power planes are on. **NOTE:** In the case where the RTC battery is dead or missing on the platform, the signal should be deasserted before the PMC_RSMRST_N signal is deasserted.
RTC_TEST_N	I	**RTC Battery Test:** An external RC circuit creates a time delay for the signal such that it will go high (to ILB_RTC_3P3_G3) sometime after the battery voltage is valid. The RC time delay should be in the 10-20 ms range. This signal will be asserted just after suspend power is up if the coin cell battery is weak. **NOTE:** This signal may also be used for debug purposes, as part of a XDP port.



19.9.2 Features

The Real Time Clock (RTC) module provides a battery backed-up date and time keeping device. Three interrupt features are available: time of day alarm with once a second to once a month range, periodic rates of 122 ms to 500 ms, and end of update cycle notification. Seconds, minutes, hours, days, day of week, month, and year are counted. The hour is represented in twelve or twenty-four hour format, and data can be represented in BCD or binary format. The design is meant to be functionally compatible with the Motorola MS146818B. The time keeping comes from a 32.768 kHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block have very specific functions. The first ten are for time and date information. The next four (0Ah to 0Dh) are registers, which configure and report RTC functions. A host-initiated write takes precedence over a hardware update in the event of a collision.

19.9.2.1 Update Cycles

An update cycle occurs once a second, if the B.SET bit is not asserted and the divide chain is properly configured. During this procedure, the stored time and date are incremented, overflow checked, a matching alarm condition is checked, and the time and date are rewritten to the RAM locations. The update cycle starts at least 488 ms after A.UIP is asserted, and the entire cycle does not take more than 1984 ms to complete. The time and date RAM locations (00h to 09h) are disconnected from the external bus during this time.

19.9.3 Interrupts

The real-time clock interrupt is internally routed within the SoC both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt does not leave the SoC, nor is it shared with any other interrupt. IRQ8# from the ILB_LPC_SERIRQ stream is ignored. However, the High Performance Event Timers can also be mapped to IRQ8#; in this case, the RTC interrupt is blocked.

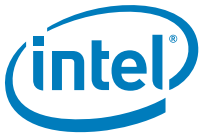
19.9.3.1 Lockable RAM Ranges

The RTC battery-backed RAM supports two 8-byte ranges that can be locked: the RC.UL and RC.LL register bits. When the locking bits are set, the corresponding range in the RAM is not readable or writable. A write cycle to those locations will have no effect. A read cycle to those locations will not return the location’s actual value (resultant value is undefined).

Once a range is locked, the range can be unlocked only by a hard reset, which will invoke the BIOS and allow it to re-lock the RAM range.

19.9.3.2 Clearing Battery-Backed RTC RAM

Clearing CMOS RAM in an SoC-based platform can be done by using a jumper on RTC_RST_N or a GPI. Implementations should not attempt to clear CMOS by using a jumper to pull RTC_VCC low.



19.9.3.2.1 Using RTC_RST_N to Clear CMOS

A jumper on RTC_RST_N can be used to clear CMOS values, as well as reset to default, the state of those configuration bits that reside in the RTC power well. When the RTC_RST_N is strapped to ground, the GEN_PMCON1.RPS register bit will be set and those configuration bits in the RTC power well will be set to their default state. BIOS can monitor the state of this bit, and manually clear the RTC CMOS array once the system is booted. The normal position would cause RTC_RST_N to be pulled up through a weak pull-up resistor. Table 106 shows which bits are set to their default state when RTC_RST_N is asserted. This RTC_RST_N jumper technique allows the jumper to be moved and then replaced—all while the system is powered off. Then, once booted, the GEN_PMCON1.RPS bit can be detected in the set state.

Table 106. Register Bits Reset by RTC_RST_N Assertion

Register Bit	Bit(s)	Default State
RCRB_GENERAL_CONTROL.TS	1	xb
GEN_PMCON1.PME_B0_S5_DIS	15	0b
GEN_PMCON1.WOL_EN_OVRD	13	0b
GEN_PMCON1.DIS_SLP_X_STRCH_SUS_UP	12	0b
GEN_PMCON1.RTC Reserved	8	0b
GEN_PMCON1.SWSMI_RATESEL	7:6	00b
GEN_PMCON1.S4MAW	5:4	00b
GEN_PMCON1.S4ASE	3	0b
GEN_PMCON1.RPS	2	1b
GEN_PMCON1.AG3E	0	0b
PM1_STS_EN.RTC_EN	26	0b
PM1_STS_EN.PWRBTNOR_STS	11	0b
PM1_CNT.SLP_TYP	12:10	0b
GPE0a_EN.PME_B0_EN	13	0b
GPE0a_EN.BATLOW_EN	10	0b

19.9.3.3 Using a GPI to Clear CMOS

A jumper on a GPI can also be used to clear CMOS values. BIOS should detect the setting of this GPI on system boot-up, and manually clear the CMOS array.

Note: The GPI strap technique to clear CMOS requires multiple steps to implement. The system is booted with the jumper in new position, then powered back down. The jumper is replaced back to the normal position, then the system is rebooted again.

Warning: Do not implement a jumper on RTC_VCC to clear CMOS.

19.9.3.4 S0ix Support

During S0i3, the RTC interface is active.

19.9.4 References

Accessing the Real Time Clock Registers and the NMI Enable Bit: http://download.intel.com/design/intarch/PAPERS/321088.pdf

19.9.5 IO Mapped Registers

The RTC internal registers and RAM is organized as two banks of 128 bytes each, called the standard and extended banks.

Note: It is not possible to disable the extended bank.

The first 14 bytes of the standard bank contain the RTC time and date information along with four registers, A - D, that are used for configuration of the RTC. The extended bank contains a full 128 bytes of battery backed SRAM. All data movement between the host CPU and the RTC is done through registers mapped to the standard I/O space.

Note: Registers reg_RTC_IR_type and reg_RTC_TR_type are used for data movement to and from the standard bank. Registers reg_RTC_RIR_type and reg_RTC_RTR_type are used for data movement to and from the extended bank. All of these registers have alias I/O locations, as indicated in Table 107.

Table 107. I/O Registers Alias Locations

Register	Original I/O Location	Alias I/O Location
reg_RTC_IR_type	70h	74h
reg_RTC_TR_type	71h	75h
reg_RTC_RIR_type	72h	76h
reg_RTC_RTR_type	73h	77h

19.9.6 Indexed Registers

The RTC contains indexed registers that are accessed via the reg_RTC_IR_type and reg_RTC_TR_type registers.

Table 108. RTC Indexed Registers (Sheet 1 of 2)

Start	End	Name
00h	00h	Seconds
01h	01h	Seconds Alarm
02h	02h	Minutes
03h	03h	Minutes Alarm
04h	04h	Hours
05h	05h	Hours Alarm
06h	06h	Day of Week

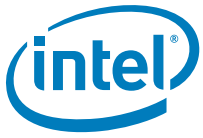
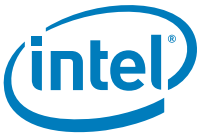


Table 108. RTC Indexed Registers (Sheet 2 of 2)

Start	End	Name
07h	07h	Day of Month
08h	08h	Month
09h	09h	Year
0Ah	0Ah	Register A
0Bh	0Bh	Register B
0Ch	0Ch	Register C
0Dh	0Dh	Register D
0Eh	7Fh	114 Bytes of User RAM



19.10 PCU - iLB - 8254 Timers

The 8254 contains three counters which have fixed uses including system timer and speaker tone. All registers are clocked by a 14.31818 MHz clock.

19.10.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

19.10.2 Features

19.10.2.1 Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is 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 one 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.

19.10.2.2 Counter 1, Refresh Request Signal

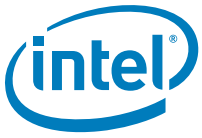
This counter is programmed for Mode 2 operation and impacts the period of the NSC.RTS register bit. Programming the counter to anything other than Mode 2 results in undefined behavior.

19.10.2.3 Counter 2, Speaker Tone

This counter provides the speaker tone and is typically programmed for Mode 3 operation. The counter provides a speaker frequency equal to the counter clock frequency (1.193 MHz) divided by the initial count value. The speaker must be enabled by a write to the NSC.SDE register bit.

19.10.2.4 S0ix Support

During S0i2 & S0i3, the 8254 timer is halted. A platform that requires the 8254 timer to be always active, should disable S0i2/3 using the S0ix_Enable register.



19.10.3 Use

19.10.3.1 Timer Programming

The counter/timers are programmed in the following fashion:

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

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

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

If a counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of all three counters. Several commands are available:

- **Control Word Command.** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command.** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command.** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

Table 109 lists the six operating modes for the interval counters.

Table 109. 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 re-triggerable 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.

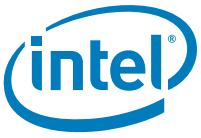


Table 109. Counter Operating Modes

Mode	Function	Description
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, etc.
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.

19.10.3.2 Reading from the Interval Timer

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

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

19.10.3.2.1 Simple Read

The first method is to perform a simple read operation. The counter is selected through Port 40h (Counter 0), 41h (Counter 1), or 42h (Counter 2).

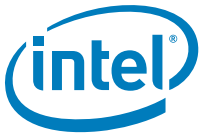
Note: Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of Counter 2, the count can be stopped by writing 0b to the NSC.TC2E register bit.

19.10.3.2.2 Counter Latch Command

The Counter Latch command, written to Port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a two-byte count. The count value is then read from each counter's Count register as was programmed by the Control register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

If a counter is latched and then, some time later, latched again before the count is read, the second Counter Latch command is ignored. The count read is the count at the time the first Counter Latch command was issued.



19.10.3.2.3 Read Back Command

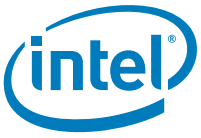
The Read Back command, written to Port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. If multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, returns the latched count. Subsequent reads return unlatched count.



19.11 PCU - iLB - High Precision Event Timer (HPET)

This function provides a set of timers that to be used by the operating system for timing events. One timer block is implemented, containing one counter and three timers.

19.11.1 Features

19.11.1.1 Non-Periodic Mode - All Timers

This mode can be thought of as creating a one-shot. When a timer is set up for non-periodic mode, it generates an interrupt when the value in the main counter matches the value in the timer's comparator register. As timers 1 and 2 are 32-bit, they will generate another interrupt when the main counter wraps.

T0CV cannot be programmed reliably by a single 64-bit write in a 32-bit environment unless only the periodic rate is being changed. If T0CV needs to be re-initialized, the following algorithm is performed:

1. Set T0C.TVS
2. Set T0CV[31:0]
3. Set T0C.TVS
4. Set T0CV[63:32]

Every timer is required to support the non-periodic mode of operation.

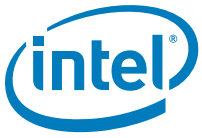
19.11.1.2 Periodic Mode - Timer 0 only

When set up for periodic mode, when the main counter value matches the value in T0CV, an interrupt is generated (if enabled). Hardware then increases T0CV by the last value written to T0CV. During run-time, T0CV can be read to find out when the next periodic interrupt will be generated. Software is expected to remember the last value written to T0CV.

Example: if the value written to T0CV is 00000123h, then

- An interrupt will be generated when the main counter reaches 00000123h.
- T0CV will then be adjusted to 00000246h.
- Another interrupt will be generated when the main counter reaches 00000246h.
- T0CV will then be adjusted to 00000369h.

When the incremented value is greater than the maximum value possible for T0CV, the value will wrap around through 0. For example, if the current value in a 32-bit timer is FFFF0000h and the last value written to this register is 20000, then after the next interrupt the value will change to 00010000h.



If software wants to change the periodic rate, it writes a new value to T0CV. When the timer's comparator matches, the new value is added to derive the next matching point. If software resets the main counter, the value in the comparator's value register must also be reset by setting T0C.TVS. To avoid race conditions, this should be done with the main counter halted. The following usage model is expected:

1. Software clears GCFG.EN to prevent any interrupts.
2. Software clears the main counter by writing a value of 00h to it.
3. Software sets T0C.TVS.
4. Software writes the new value in T0CV.
5. Software sets GCFG.EN to enable interrupts.

19.11.1.2.1 Interrupts

If each timer has a unique interrupt and the timer has been configured for edge-triggered mode, then there are no specific steps required. If configured to level-triggered mode, then its interrupt must be cleared by software by writing a '1' back to the bit position for the interrupt to be cleared.

Interrupts associated with the various timers have several interrupt mapping options. Software should mask GCFG.LRE when reprogramming HPET interrupt routing to avoid spurious interrupts.

19.11.1.2.2 Mapping Option #1: Legacy Option (GCFG.LRE set)

This forces the following mapping:

Table 110. 8254 Interrupt Mapping

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	The 8254 timer will not cause any interrupts
1	IRQ8	IRQ8	RTC will not cause any interrupts.
2	T2C.IR	T2C.IRC	

19.11.1.2.3 Mapping Option #2: Standard Option (GCFG.LRE cleared)

Each timer has its own routing control. The interrupts can be routed to various interrupts in the I/O APIC. T[2:0]C.IRC indicates which interrupts are valid options for routing. If a timer is set for edge-triggered mode, the timers should not be shared with any other interrupts.

19.11.1.3 S0ix Support

During S0i1, the HPET is kept running. During S0i2 & S0i3, the HPET is halted.

19.11.1.4 S0ix Support

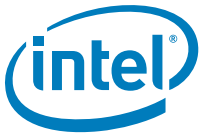
Prior to entry into S0i2 or S0i3 state, the driver/OS must set HPET_GCFG.EN to 0b to indicate $RTD3_{hot}$ status.

19.11.2 References

IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a: http://www.intel.com/hardware design/hpetspec_1.pdf

19.11.3 Memory Mapped Registers

The register space is memory mapped to a 1K block at address FED00000h. All registers are in the core well. Accesses that cross register boundaries result in undefined behavior.



19.12 PCU - iLB - GPIO

187 GPIOs are available for use. Most of these GPIOs can be used as legacy GPIOs. This chapter describes their use as legacy GPIOs.

19.12.1 Signal Descriptions

Refer Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 20, "Electrical Specifications"
- **Description**: A brief explanation of the signal's function

19.12.2 Features

GPIOs can generate general purpose events (GPEs) on rising and/or falling edges.

19.12.2.1 GPIO controller

The GPIO controllers handle all GPIO interface to SoC,

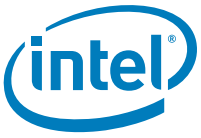
- GPIO NORTH - used for Camera sensors, DFX, SVID, and Display Pins.
- GPIO SOUTHEAST - Defines the pads/Pins for MMC/SD host controller, LPC pins, FAST SPI pins and Platform Clock.
- GPIO SOUTHWEST - Defines the Pads/Pins for HS UART,I2S HS, LPE, PCIe and SPI pins.
- GPIO EAST - Defines the Pads/Pins for SoC power state related signals of PMU and ISH pins.

19.12.3 Use

Each GPIO has six registers that control how it is used, or report its status:

- Use Select
- I/O Select
- GPIO Level
- Trigger Positive Edge
- Trigger Negative Edge
- Trigger Status

The Use Select register selects a GPIO pin as a GPIO, or leaves it as its programmed function. This register must be set for all other registers to affect the GPIO.



The I/O Select register determines the direction of the GPIO.

The Trigger Positive Edge and Trigger Negative Edge registers enable general purpose events on a rising and falling edge respectively. This only applies to GPIOs set as input.

The Trigger Status register is used by software to determine if the GPIO triggered a GPE. This only applies to GPIOs set as input and with one or both of the Trigger modes enabled.

Additionally, there is one additional register for each S5 GPIO:

- Wake Enable

This register allows S5 GPIOs to trigger a wake event based on the Trigger registers' settings.

19.12.4 GPIO Registers

19.12.4.1 SD Card and LPC Pins (3.3V versus 1.8V Modes)

The CFIO cells for both the SD Card Pins (SDMMC3_*) and LPC (LPC_*) are 3.3V capable.

To use as 1.8V IOs:

• Set power supply to 1.8V for the pads.

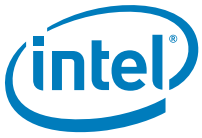
• Set v1p8mode in family configuration register.

• Trigger a RCOMP cycle using Family RCOMP register

• Copy RCOMP value to Family p and n strength values.

Note: All GPIO registers must be accessed as double words. Unpredictable results will occur otherwise.

Note: All MMIO GPIO *_PAD_VAL's must set Ienenb = 0 in order to read the pad_val of the GPIO. This applies to RO GPIO's as well.



19.13 PCU - iLB - Interrupt Decoding and Routing

The interrupt decoder is responsible for receiving interrupt messages from other devices in the SoC and decoding them for consumption by the interrupt router, the "PCU - iLB - 8259 Programmable Interrupt Controllers (PIC)" and/or the "PCU - iLB - IO APIC".

The interrupt router is responsible for mapping each incoming interrupt to the appropriate PIRQx, for consumption by the "PCU - iLB - 8259 Programmable Interrupt Controllers (PIC)" and/or the "PCU - iLB - IO APIC".

19.13.1 Features

19.13.1.1 Interrupt Decoder

The interrupt decoder receives interrupt messages from devices in the SoC. These interrupts can be split into two primary groups:

- For consumption by the interrupt router
- For consumption by the 8259 PIC

19.13.1.1.1 For Consumption by the Interrupt Router

When a PCI-mapped device in the SoC asserts or de-asserts an INT[A:D] interrupt, an interrupt message is sent to the decoder. This message is decoded to indicate to the interrupt router which specific interrupt is asserted or de-asserted and which device the INT[A:D] interrupt originated from.

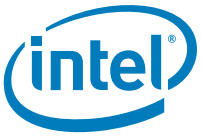
19.13.1.1.2 For Consumption by the 8259 PIC

When a device in the SoC asserts or de-asserts a legacy interrupt (IRQ), an interrupt message is sent to the decoder. This message is decoded to indicate to the 8259 PIC, which specific interrupt (IRQ[3, 4, 14 or 15]) was asserted or de-asserted.

19.13.1.2 Interrupt Router

The interrupt router aggregates the INT[A:D] interrupts for each PCI-mapped device in the SoC, received from the interrupt decoder, and the INT[A:D] interrupts direct from the Serialized IRQ controller. It then maps these aggregated interrupts to 8 PCI based interrupts: PIRQ[A:H]. This mapping is configured using the IR[31:0] registers.

PCI based interrupts PIRQ[A:H] are then available for consumption by either the 8259 PICs or the IO-APIC, depending on the configuration of the 8 PIRQx Routing Control Registers: PIRQA, PIQRB, PIRQC, PIRQD, PIRQE, PIRQF, PIRQG, PIRQH.



19.13.1.2.1 Routing PCI Based Interrupts to 8259 PIC

The interrupt router can be programmed to allow PIRQA-PIRQH to be routed internally to the 8259 as ISA compatible interrupts IRQ 3–7, 9–12 & 14–15. The assignment is programmable through the 8 PIRQx Routing Control Registers: PIRQA, PIQRB, PIRQC, PIRQD, PIRQE, PIRQF, PIRQG, PIRQH. One or more PIRQs can be routed to the same IRQ input. If ISA Compatible Interrupts are not required, the Route registers can be programmed to disable steering.

The PIRQx# lines are defined as active low, level sensitive. When a PIRQx# is routed to specified IRQ line, software must change the IRQ's corresponding ELCR bit to level sensitive mode. The SoC internally inverts the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an active high device (through SERIRQ). However, active low interrupts can share their interrupt with PCI interrupts.

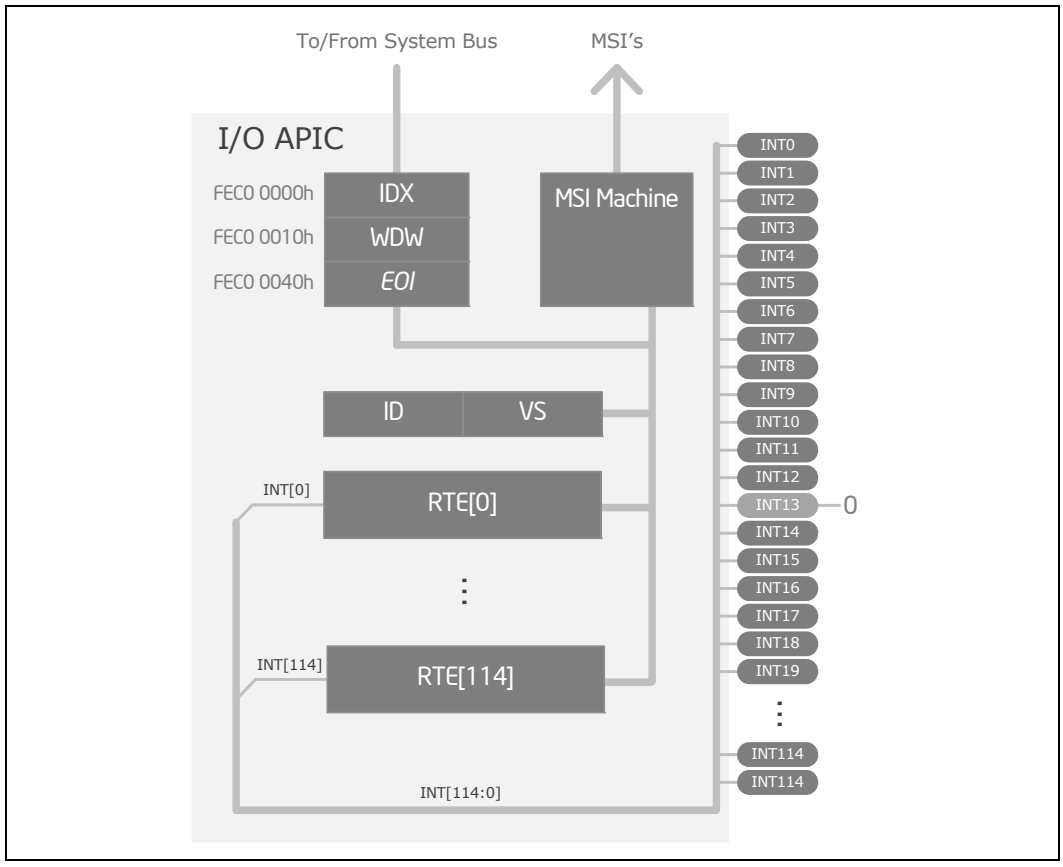
19.14 PCU - iLB - IO APIC

The IO Advanced Programmable Interrupt Controller (APIC) is used to support line interrupts more flexibly than the 8259 PIC. Line interrupts are routed to it from multiple sources, including legacy devices, via the interrupt decoder and serial IRQs, or they are routed to it from the interrupt router in the iLB. These line based interrupts are then used to generate interrupt messages targeting the local APIC in the processor.

19.14.1 Features

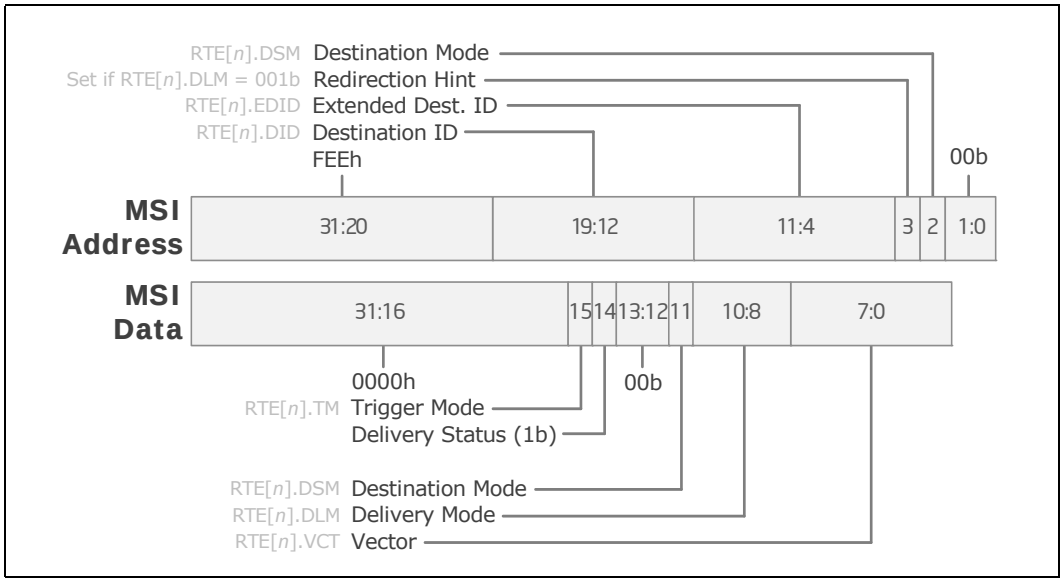
- 115 interrupt lines
  - — IRQ0-114
- Edge or level trigger mode per interrupt
- Active low or high polarity per interrupt
- Works with local APIC in processor via MSIs
- MSIs can target specific processor core
- Established APIC programming model

Figure 33. Detailed Block Diagram



MSIs generated by the I/O APIC are sent as 32-bit memory writes to the Local APIC. The address and data of the write transaction are used as follows.

Figure 34. MSI Address and Data



Destination ID (DID) and Extended Destination ID (EDID) are used to target a specific processor core's local APIC.

19.14.2 Use

The I/O APIC contains indirectly accessed I/O APIC registers and normal memory mapped registers. There are three memory mapped registers:

- Index Register (IDX)
- Window Register (WDW)
- End Of Interrupt Register (EOI)

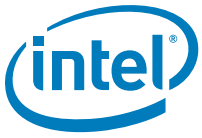
The Index register selects an indirect I/O APIC register (ID/VS/RTE[*n*]) to appear in the Window register.

The Window register is used to read or write the indirect register selected by the Index register.

The EOI register is written to by the Local APIC in the processor. The I/O APIC compares the lower eight bits written to the EOI register to the Vector set for each interrupt (RTE.VCT). All interrupts that match this vector will have their RTE.RIRR register cleared. All other EOI register bits are ignored.

19.14.3 References

TBD



19.14.4 Memory Mapped Registers

TBD

19.14.5 Indirect I/O APIC Registers

These registers are selected with the IDX register, and read/written through the WDW register. Accessing these registers must be done as DW requests, otherwise unspecified behavior will result. Software should not attempt to write to reserved registers. Reserved registers may return non-zero values when read.

Note: There is one pair of redirection (RTE) registers per interrupt line. Each pair forms a 64-bit RTE register.

Note: Specified offsets should be placed in IDX, not added to IDX.

19.15 PCU - iLB - 8259 Programmable Interrupt Controllers (PIC)

The SoC provides an ISA-compatible programmable interrupt controller (PIC) that incorporates the functionality of two, cascaded 8259 interrupt controllers.

19.15.1 Features

In addition to providing support for ISA compatible interrupts, this interrupt controller can also support PCI based interrupts (PIRQs) by mapping the PCI interrupt onto a compatible ISA interrupt line. Each 8259 controller supports eight interrupts, numbered 0–7. Table 111 shows how the controllers are connected.

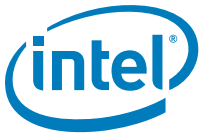
Note: The SoC does not implement any external PIRQ# signals. The PIRQs referred to in this chapter originate from the interrupt routing unit.

Table 111. Interrupt Controller Connections

8259	8259 Input	Connected Pin / Function
Master	0	Internal Timer / Counter 0 output or HPET #0; determined by GCFG.LRE register bit
	1	IRQ1 using SERIRQ, Keyboard Emulation
	2	Slave controller INTR output
	3	IRQ3 via SERIRQ, PIRQx or PCU UART 1
	4	IRQ4 via SERIRQ or PIRQx
	5	IRQ5 via SERIRQ or PIRQx
	6	IRQ6 via SERIRQ or PIRQx
	7	IRQ7 via SERIRQ or PIRQx
Slave	0	Inverted IRQ8# from internal RTC or HPET
	1	IRQ9 via SERIRQ, SCI or PIRQx
	2	IRQ10 via SERIRQ, SCI or PIRQx
	3	IRQ11 via SERIRQ, SCI, HPET or PIRQx
	4	IRQ12 via SERIRQ, PIRQx or mouse emulation
	5	None
	6	PIRQx
	7	IRQ15 via SERIRQ or PIRQx o

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

Interrupts can be programmed individually to be edge or level, except for IRQ0, IRQ2 and IRQ8#.



Note: Active-low interrupt sources (such as a PIRQ#) are inverted inside the SoC. In the following descriptions of the 8259s, the interrupt levels are in reference to the signals at the internal interface of the 8259s, after the required inversions have occurred. Therefore, the term "high" indicates "active," which means "low" on an originating PIRQ#.

19.15.1.1 Interrupt Handling

19.15.1.1.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 112 defines the IRR, ISR, and IMR.

Table 112. Interrupt Status Registers

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

19.15.1.1.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle that is translated into a Interrupt Acknowledge Cycle to the SoC. The PIC translates this command into two internal INTA# pulses expected by the 8259 controllers. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave sends the interrupt vector to the processor with the acknowledged interrupt code. This code is based upon the ICW2.IVBA bits, combined with the ICW2.IRL bits representing the interrupt within that controller.

Note: References to ICWx and OCWx registers are relevant to both the master and slave 8259 controllers.

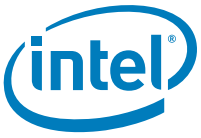


Table 113. Content of Interrupt Vector Byte

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

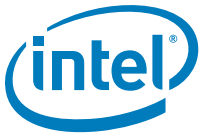
19.15.1.1.3 Hardware/Software Interrupt Sequence

1. One or more of the Interrupt Request lines (IRQ) are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor if an asserted interrupt is not masked.
3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle.
4. Upon observing the special cycle, the SoC converts it into the two cycles that the internal 8259 pair can respond to. Each cycle appears as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.
5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine if it must respond with an interrupt vector during the second INTA# pulse.
6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. If no interrupt request is present because the request was too short in duration, the PIC returns vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

19.15.1.2 Initialization Command Words (ICWx)

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

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



19.15.1.2.1 ICW1

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

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

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

19.15.1.2.2 ICW2

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

19.15.1.2.3 ICW3

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

- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. Within the SoC, IRQ2 is used. Therefore, MICW3.CCC is set to a 1, and the other bits are set to 0s.
- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller if the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and if it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

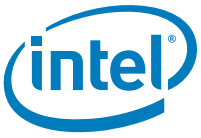
19.15.1.2.4 ICW4

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

19.15.1.3 Operation Command Words (OCW)

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

- OCW1 masks and unmasks interrupt lines.



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

19.15.1.4 Modes of Operation

19.15.1.4.1 Fully Nested Mode

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

Interrupt priorities can be changed in the rotating priority mode.

19.15.1.4.2 Special Fully-Nested Mode

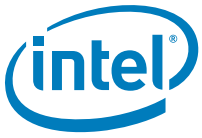
This mode is used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully-nested mode is programmed to the master controller. This mode is similar to the fully-nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave are recognized by the master and initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.
- When exiting the Interrupt Service routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a Non-Specific EOI command to the slave and then reading its ISR. If it is 0, a non-specific EOI can also be sent to the master.

19.15.1.4.3 Automatic Rotation Mode (Equal Priority Devices)

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

There are two ways to accomplish automatic rotation using OCW2.REOI; the Rotation on Non-Specific EOI Command (OCW2.REOI=101b) and the rotate in automatic EOI mode which is set by (OCW2.REOI=100b).



19.15.1.4.4 Specific Rotation Mode (Specific Priority)

Software can change interrupt priorities by programming the bottom priority. For example, if IRQ5 is programmed as the bottom priority device, then IRQ6 is the highest priority device. The Set Priority Command is issued in OCW2 to accomplish this, where: OCW2.REOI=11xb, and OCW2.ILS is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes can be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (OCW2.REOI=111b) and OCW2.ILS=IRQ level to receive bottom priority.

19.15.1.4.5 Poll Mode

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

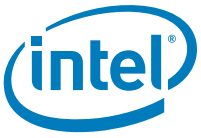
The Poll command is issued by setting OCW3.PMC. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit if there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read contains a 1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

19.15.1.4.6 Edge and Level Triggered Mode

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

If an ELCR bit is 0, an interrupt request will be recognized by a low-to-high transition on the corresponding IRQ input. The IRQ input can remain high without generating another interrupt. If an ELCR bit is 1, an interrupt request will be recognized by a high level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

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



19.15.1.5 End of Interrupt (EOI) Operations

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

19.15.1.5.1 Normal End of Interrupt

In normal EOI, software writes an EOI command before leaving the interrupt service routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific. When a Non-Specific EOI command is issued, the PIC clears the highest ISR bit of those that are set to 1. Non-Specific EOI is the normal mode of operation of the PIC within the SoC, as the interrupt being serviced currently is the interrupt entered with the interrupt acknowledge. When the PIC is operated in modes that preserve the fully nested structure, software can determine which ISR bit to clear by issuing a Specific EOI.

An ISR bit that is masked is not cleared by a Non-Specific EOI if the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

19.15.1.5.2 Automatic End of Interrupt Mode

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

Note: Both the master and slave PICs have an AEOI bit: MICW4.AEOI and SICW4.AEOI respectively. Only the MICW4.AEOI bit should be set by software. The SICW4.AEOI bit should not be set by software.

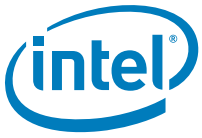
19.15.1.6 Masking Interrupts

19.15.1.6.1 Masking on an Individual Interrupt Request

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

19.15.1.6.2 Special Mask Mode

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



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

The special mask mode is set by OCW3.ESMM=1b & OCW3.SMM=1b, and cleared where OCW3.ESMM=1b & OCW3.SMM=0b.

19.15.1.7 S0ix Support

During S0i2 & S0i3, the 8259 PICs are disabled. A platform that requires the 8259 PICs to be always active, should disable S0i2/3 using the S0ix_Enable register.

19.15.2 IO Mapped Registers

The interrupt controller registers are located at 20h and 21h for the master controller (IRQ0 - 7), and at A0h and A1h for the slave controller (IRQ8 - 13). These registers have multiple functions, depending upon the data written to them. Table 114 is a description of the different register possibilities for each address.

Note: The register descriptions after Table 114 represent one register possibility.

Table 114. I/O Registers Alias Locations

Registers	Original I/O Location	Alias I/O Locations
MICW1 MOCW2 MOCW3	20h	24h
		28h
		2Ch
		30h
		34h
		38h
		3Ch
MICW2 MICW3 MICW4 MOCW1	21h	25h
		29h
		2Dh
		31h
		35h
		39h
		3Dh

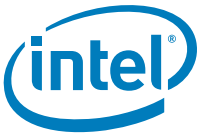
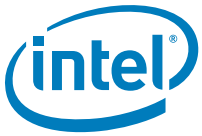


Table 114. I/O Registers Alias Locations

Registers	Original I/O Location	Alias I/O Locations
SICW1 SOCW2 SOCW3	A0h	A4h
		A8h
		ACh
		B0h
		B4h
		B8h
		BCh
SICW2 SICW3 SICW4 SOCW1	A1h	A5h
		A9h
		ADh
		B1h
		B5h
		B9h
		BDh
ELCR1	4D0h	N/A
ELCR2	4D1h	N/A

§



20 Electrical Specifications

This chapter is categorized into the following sections:

- "Absolute Maximum and Minimum Specifications"
- "Thermal Specifications"
- "Storage Conditions"
- "Voltage and Current Specifications"
- "Crystal Specifications"
- "DC Specifications"
- "AC Specifications"

20.1 Absolute Maximum and Minimum Specifications

The absolute maximum and minimum specifications are used to specify conditions allowable outside of the functional limits of the SoC, but with possible reduced life expectancy once returned to function limits.

At conditions exceeding absolute specifications, neither functionality nor long term reliability can be expected. Parts may not function at all once returned to functional limits.

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

20.2 Thermal Specifications

These specifications define the operating thermal limits of the SoC. Thermal solutions not designed to provide the following level of thermal capability may affect the long-term reliability of the processor and system, but more likely result in performance throttling to ensure silicon junction temperatures within spec. For more details on thermal solution design, refer to this product's Thermal/Mechanical Design Guide.

This section specifies the thermal specifications for all SKUs. Some definitions are needed, however. "Tj Max" defines the maximum operating silicon junction temperature. Unless otherwise specified, all specifications in this document assume Tj Max as the worse case junction temperature. This is the temperature needed to ensure TDP specifications when running at guaranteed CPU and graphics frequencies. "TDP" defines the thermal dissipated power for a worse case estimated real world thermal scenario. "SDP", or scenario dissipated power, defines the thermal dissipated power under a lighter workload specific to a user scenario and at a lower thermal junction

temperature than Tj Max. Note that turbo frequencies are opportunistically selected when thermal headroom exists. Automatic throttling along with a proper thermal solution ensure Tj Max will not be exceeded.

Table 115. Thermal Specifications

	MSP-T4	VMS-T3	Co-POP	MSP-T4 Refresh
T_j Max	105 °C	105 °C	90 °C	105 °C
T_j Min	0 °C	0 °C	0 °C	0 °C
T_j @ Max. Steady State Power	85 °C	85 °C	85 °C	85 °C
SDP	2W	2.2W	2W	2W

20.3 Storage Conditions

This section specifies absolute maximum and minimum storage temperature and humidity limits for given time durations. Failure to adhere to the specified limits could result in physical damage to the component. If this is suspected, Intel recommends a visual inspection to determine possible physical damage to the silicon or surface components.

Table 116. Storage Conditions Prior to Board Attach

Symbol	Parameter	Min	Max
Tabsolute storage	Device storage temperature when exceeded for any length of time.	-25 °C	125 °C
Tshort term storage	The ambient storage temperature and time for up to 72 hours.	-25 °C	85 °C
Tsustained storage	The ambient storage temperature and time for up to 30 months.	5 °C	40 °C
RHsustained storage	The maximum device storage relative humidity for up to 30 months.		60% @ 24 °C

Note:

- Specified temperatures are not to exceed values based on data collected. Exceptions for surface mount re-flow are specified by the applicable JEDEC standard. Non-adherence may affect processor reliability.
- Component product device storage temperature qualification methods may follow JESD22-A119 (low temperature) and JESD22-A103 (high temperature) standards when applicable for volatile memory.
- Component stress testing is conducted in conformance with JESD22-A104.
- The JEDEC J-JSTD-020 moisture level rating and associated handling practices apply to all moisture sensitive devices removed from the moisture barrier bag.

20.3.1 Post Board-Attach

The storage condition limits for the component once attached to the application board are not specified. Intel does not conduct component-level certification assessments post board-attach given the multitude of attach methods, socket types, and board types used by customers.

Provided as general guidance only, board-level Intel-branded products are specified and certified to meet the following temperature and humidity limits:

- Non-Operating Temperature Limit: -40 °C to 70 °C
- Humidity: 50% to 90%, non-condensing with a maximum wet-bulb of 28 °C

20.4 Voltage and Current Specifications

The I/O buffer supply voltages are specified at the SoC package balls. The tolerances shown in Table 131 are inclusive of all noise from DC up to 20 MHz. The voltage rails should be measured with a bandwidth limited oscilloscope with a roll-off of 3 dB/decade above 20 MHz under all operating conditions. Table 118 indicates which supplies are connected directly to a voltage regulator or to a filtered voltage rail. For voltage rails that are connected to a filter, they should be measured at the input of the filter. If the recommended platform decoupling guidelines cannot be met, the system designer will have to make trade-offs between the voltage regulator out DC tolerance and the decoupling performances of the capacitor network to stay within the voltage tolerances listed below.

Note: The SoC is a pre-launch product. Voltage and current specifications are subject to change.

Table 117. SoC Power Rail DC Specs and Max Current

Platform Rail		Voltage Tolerances	Max Icc
V1P05A	UNCORE1_V1P05A_G3	1.05 V DC: ±2% AC: ±2%	1700 mA
	UNCORE2_V1P05A_G3		
	DDR_V1P05A_G3		
	USB3_V1P05A_G3		
	USBSSIC_V1P05A_G3		
	F_V1P05A_G3		
	PCIECLK_V1P05A_G3		
V1P15	CORE_V1P15_S0iX	1.15 V DC: ±2% AC: ±3%	2100 mA
	DDI_V1P15_S0iX		
	UNCORE_V1P15_S0iX		
	F_V1P15_S0iX		

Table 117. SoC Power Rail DC Specs and Max Current

Platform Rail		Voltage Tolerances	Max Icc
V1P2A	USBSSIC_V1P2A_G3	1.24 V DC: ±2% AC: ±2%	67 mA
	MIPI_V1P2A_G3		
	USBHSIC_V1P2A_G3		
V1P8A	USB_V1P8A_G3	1.8 V DC: ±2% AC: ±2%	971 mA
	UNCORE_V1P8A_G3		
	GPIOSE_V1P8A_G3		
	GPION_V1P8A_G3		
	F_V1P8A_G3		
V3P3A	USB_V3P3A_G3	3.3 V DC: ±2% AC: ±2%	196 mA
	F_V3P3A_G3		
	RTC_V3P3A_G5		
V3P3A_V1P8A	SDIO_V3P3A_V1P8A_G3	1.8 V/3.3 V DC: ±2% AC: ±2%	-
	LPC_V3P3A_V1P8A_S4		
VSFR	ICLK_VSFR_G3	1.05 V/1.24 V/ 1.35 V DC: ±2% AC: ±3%	-
	CORE0_VSFR_G3		
	CORE1_VSFR_G3		
	UNCORE_VSFR_G3		
VCC0	CORE_VCC0_S0iX	See Table 118	3200 mA
	CORE_VCC0_SENSE		
VCC1	CORE_VCC1_S0iX	See Table 118	3200 mA
	CORE_VCC1_SENSE		
VNN	UNCORE_VNN_S4	See Table 118	2500 mA
	UNCORE_VNN_SENSE		
VGG	DDI_VGG_S0iX	See Table 118	8000 mA
	DDI_VGG_SENSE		

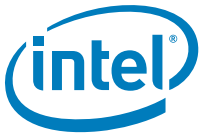


Table 117. SoC Power Rail DC Specs and Max Current

Platform Rail		Voltage Tolerances	Max Icc
VDDQ	DDI1_VDDQ_G3	1.24 V/1.35 V DC: ±2% AC: ±2%	1900 - 2500mA
	DDI2_VDDQ_G3		
	USB_VDDQ_G3		
VDDQG	DDR_VDDQG_S4	1.24 V/1.35 V DC: ±2% AC: ±2%	
	DDRCH0_VDDQG_S4		
	DDRCH1_VDDQG_S4		
	DDRSFRCH0_VDDQG_S4		
	DDRSFRCH1_VDDQG_S4		
V3P3RTC	RTC_V3P3RTC_G5	G5: 2-3 V at battery Otherwise V3P3A (pre diode drop)	-

Note: RTC_VCC average current draw (G5) is specified at 27°C under battery conditions

20.4.1 VCC and VNN Voltage Specifications

Table 118 and Table 131 list the DC specifications for the SoC power rails. They are valid only while meeting specifications for junction temperature, clock frequency, and input voltages. Care should be taken to read all notes associated with each parameter.

Table 118. VCC and VNN DC Voltage Specifications

Symbol	Parameter	Min	Typ	Max	Unit	Note
CORE_VCC VID	Core VID Target Range	0.75		1.15	V	
CORE_VCC0_S0iX	V_{CC0} for SoC Core 0	See VCC VID			V	2, 5
CORE_VCC1_S0iX	V_{CC1} for SoC Core 1	See VCC VID			V	2, 5
UNCORE_VNN VID	Uncore VID Target Range	0.75		1.15	V	
UNCORE_VNN_S4	V_{NN} for SoC Uncore	See VNN VID			V	2, 5
DDI_VGG_S0iX	V_{GG} for SoC Display	0.75		1.15	V	
CORE_VCC/ UNCORE_VNN V_{BOOT}	Default target V_{CC}/V_{NN} voltage for initial power up.		1.0 or 1.1		V	4
VCC0/1 Tolerance	Tolerance of VCC0/1 voltage at VID target.	DC: ±2% AC: ±3%			%	
VNN Tolerance	Tolerance of VNN voltage at VID target.	DC: ±2% AC: ±2%			%	
VGG Tolerance	Tolerance of VGG voltage at VID target.	DC: ±2% AC: ±3%			%	

NOTES:
1. See Intel representative for load line and tolerance details.
2. Each SoC is programmed with voltage identification value (VID), which is set at manufacturing and cannot be altered. Individual VID values are calibrated during manufacturing such that two SoCs at the same frequency may have different settings within the VID range. Note this differs from the VID employed by the SoC during a power management event.
3. These are pre-silicon estimates and are subject to change.
4. See the VR12/IMVP7 Pulse Width Modulation specification for additional details. Either value is ok.

20.5 Crystal Specifications

There are two crystal oscillators. One for RTC which maintains time and provides initial timing reference for power sequencing. The other is for the Integrated Clock, which covers clocking for the entire SoC.

Table 119. ILB RTC Crystal Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
F_{RTC}	Frequency	-	32.768	-	kHz	1
T_{PPM}	Crystal frequency tolerance (see notes)	-	-	+/-50	ppm	1
R_{ESR}	ESR	-	-	50	kOhm	1
$C_{X1,2}$	Capacitance of X1, X2 pins				pF	1

NOTES:
1. These are the specifications needed to select a crystal oscillator for the RTC circuit.
2. Crystal tolerance impacts RTC time. A 10 ppm crystal is recommended for 1.7 s tolerance per day, RTC circuit itself contributes addition 10 ppm for a total of 20 ppm in this example.

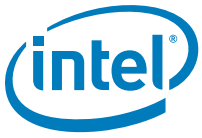
Table 120. Integrated Clock Crystal Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
F_{ICLK}	Frequency	-	19.2	-	MHz	1
T_{PPM}	Crystal frequency tolerance & stability	-	-	+/-100	ppm	1
P_{DRIVE}	Crystal drive load	-	-	100	uW	1
R_{ESR}	ESR	-	-	100	Ohm	1
C_{LOAD}	Crystal load capacitance		18		pF	
C_{SHUNT}	Crystal shunt capacitance	-	-	6	pF	1
$C_{IN/OUT}$	Capacitance of oscillator pins				pF	1

NOTE: These are the specifications needed to select a crystal oscillator for the Integrated Clock circuit. Crystal must be AT cut, fundamental, parallel resonance.

20.6 DC Specifications

Platform reference voltages are specified at DC only. V_{REF} measurements should be made with respect to the supply voltages specified in "Voltage and Current Specifications".



Note: $V_{IH/OH}$ Max and $V_{IL/OL}$ Min values are bounded by reference voltages.

See the following DC Specifications in this section:

- "Display DC Specification"
- "MIPI-Camera Serial Interface (CSI) DC Specification"
- "SDIO DC Specification"
- "SD Card DC Specification"
- "eMMC 4.51 DC Specification"
- "JTAG DC Specification"
- "DDR3L-RS Memory Controller DC Specification"
- "LPDDR3 Memory Controller DC Specification"
- "USB 2.0 Host DC Specification"
- "USB 3.0 DC Specification"
- "LPC DC Specification"
- "SPI DC Specification"
- "Power Management/Thermal (PMC) & RTC DC Specification"
- "SVID DC Specification"
- "GPIO DC Specification"
- "SIO - I^2C DC Specification"
- "SIO - UART DC Specification"
- "I^2S (Audio) DC Specification"
- "PCI Express DC Specification"

Note: Care should be taken to read all notes associated with each parameter.

20.6.1 Display DC Specification

DC specifications for display interfaces:

- "Display Port DC Specification"
- "HDMI DC Specification"
- "Embedded Display Port DC Specification"
- "Display Port AUX Channel DC Specification"
- "Embedded Display Port AUX Channel DC Specification"
- "DDC Signal DC Specification"
- "MIPI DSI DC Specification"

20.6.1.1 Display Port DC Specification

Table 121. Display Port DC specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
$V_{TX-DIFFp-p-Level0}$	Differential Peak-to-peak Output Voltage Level 0	0.34	0.4	0.46	V	
$V_{TX-DIFFp-p-Level1}$	Differential Peak-to-peak Output Voltage Level 1	0.51	0.6	0.68	V	
$V_{TX-DIFFp-p-Level2}$	Differential Peak-to-peak Output Voltage Level 2	0.69	0.8	0.92	V	
$VT_{X-DIFFp-p-Level3}$	Differential Peak-to-peak Output Voltage Level 3	0.85	1.2	1.38	V	
$V_{TX-PREEMP-RATIO}$	No Pre-emphasis	0.0	0.0	0.0	dB	
	3.5 dB Pre-emphasis	2.8	3.5	4.2	dB	
	6.0 dB Pre-emphasis	4.8	6.0	7.2	dB	
	9.5 dB Pre-emphasis	7.5	9.5	11.4	dB	
$V_{TX-DC-CM}$	Tx DC Common Mode Voltage	0		2.0	V	
$RL_{TX-DIFF}$	Differential Return Loss at 0.675GHz at Tx Package pins	12			dB	
	Differential Return Loss at 1.35 GHz at Tx Package pins	9			dB	1
C_{TX}	TX Output Capacitance			1.5	pF	2

NOTES:
1. Straight loss line between 0.675 GHz and 1.35 GHz
2. Represents only the effective lump capacitance seen at the SoC interface that shunts the TX termination.

20.6.1.2 HDMI DC Specification

Table 122. HDMI DC specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
Voff	Single Ended Standby (off), output voltage	-10		10	mV	1 @ AVcc
Vswing	Single Ended output swing voltage	400		600	mV	
V_{OH} (<=165 MHz)	Single Ended high level, output voltage	-10		10	mv	1 @ AVcc

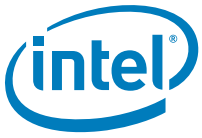


Table 122. HDMI DC specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{OH}(>165 MHz)	Single Ended high level, output voltage	-200		10	mV	1 @ AVcc
V_{OL}(<=165 MHz)	Single Ended low level, output voltage	-600		-400	mV	1 @ AVcc
V_{OL}(>165MHz)	Single Ended low level, output voltage	-700		-400	mV	1 @ AVcc

NOTE: 1. The min/max values are with reference to AVcc =Analog Voltage level

20.6.1.3 Embedded Display Port DC Specification

Table 123. Embedded Display Port DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
$V_{TX-DIFFp-p-Level0}$	Differential Peak-to-peak Output Voltage Level 0	0.18	0.2	0.22	V	1,2
$V_{TX-DIFFp-p-Level1}$	Differential Peak-to-peak Output Voltage Level 1	0.2	0.25	0.275	V	1,2
$V_{TX-DIFFp-p-Level2}$	Differential Peak-to-peak Output Voltage Level 2	0.27	0.3	0.33	V	1,2
$VT_{X-DIFFp-p-Level3}$	Differential Peak-to-peak Output Voltage Level 3	0.315	0.35	0.385	V	1,2
$VT_{X-DIFFp-p-Level4}$	Differential Peak-to-peak Output Voltage Level 4	0.36	0.4	0.44	V	1,2
$VT_{X-DIFFp-p-Level5}$	Differential Peak-to-peak Output Voltage Level 5	0.405	0.45	0.495	V	1,2
$VT_{X-DIFFp-p-MAX}$	Maximum Allowed Differential Peak-to-peak Output Voltage			1.380	V	3
$V_{TX-DC-CM}$	Tx DC Common Mode Voltage	0		2.0	V	1
$V_{TX-PREEMP-RATIO}$	No Pre-emphasis	0.0	0.0	0.0	dB	1
	3.5 dB Pre-emphasis	2.8	3.5	4.2	dB	1
	6.0 dB Pre-emphasis	4.8	6.0	7.2	dB	1
	9.5 dB Pre-emphasis	7.5	9.5	11.4	dB	1
$RL_{TX-DIFF}$	Differential Return Loss at 0.675GHz at Tx Package pins	12			dB	4
	Differential Return Loss at 1.35 GHz at Tx Package pins	9			dB	4
C_{TX}	TX Output Capacitance			1.5	pF	5

NOTES:
1. Steps between VTX-DIFFP-P voltages must be monotonic. The actual VTX-DIFFP-P-1 voltage must be equal to or greater than the actual VTX-DIFFP-P-0 voltage; the actual VTX-DIFFP-P-2 voltage must be greater than the actual VTX-DIFFP-P-1 voltage; etc.
2. The recommended minimum VTX-DIFFP-P delta between adjacent voltages is mV.
3. Allows eDP Source devices to support differential signal voltages compatible with eDP v1.3 (and lower) devices and designs.
4. Straight loss line between 0.675 GHz and 1.35 GHz.
5. Represents only the effective lump capacitance seen at the SoC interface that shunts the TX termination.

20.6.1.4 Display Port AUX Channel DC Specification

Table 124. DDI AUX Channel DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
$V_{AUX-DIFFp-p}$	AUX Peak-to-peak Voltage at a transmitting Device	0.29		1.38	V	1
$V_{AUX-\_TERM\_R}$	AUX CH termination DC resistance		100		Ω	
$V_{AUX-DC-CM}$	AUX DC Common Mode Voltage	0		2.0	V	2
$V_{AUX-TURN-CM}$	AUX turn around common mode voltage			0.3	V	3
I_{AUX_SHORT}	AUX Short Circuit Current Limit			90	mA	4
C_{AUX}	AC Coupling Capacitor	75		200	nF	5

NOTES:
1. $V_{AUX-DIFFp-p} = 2*|V_{AUXP} - V_{AUXM}|$
2. Common mode voltage is equal to V_{bias_Tx} (or V_{bias_Rx}) voltage.
3. Steady state common mode voltage shift between transmit and receive modes of operation.
4. Total drive current of the transmitter when it is shorted to its ground.
5. All DisplayPort Main Link lanes as well as AUX CH must be AC coupled. AC coupling capacitors must be placed on the transmitter side. Placement of AC coupling capacitors on the receiver side is optional.

20.6.1.5 Embedded Display Port AUX Channel DC Specification

Table 125. Embedded Display Port AUX Channel DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
$V_{AUX-DIFFp-p}$	AUX Peak-to-peak Voltage at a transmitting Device	0.29		1.38	V	1
$V_{AUX-\_TERM\_R}$	AUX CH termination DC resistance		100		Ω	
$V_{AUX-DC-CM}$	AUX DC Common Mode Voltage	0		1.2	V	2

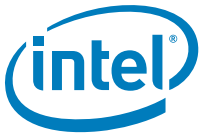


Table 125. Embedded Display Port AUX Channel DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
$V_{AUX-TURN-CM}$	AUX turn around common mode voltage			0.3	V	3
I_{AUX_SHORT}	AUX Short Circuit Current Limit			90	mA	4
C_{AUX}	AC Coupling Capacitor	75		200	nF	5

NOTES:
1. $V_{AUX-DIFFp-p} = 2*|V_{AUXP} - V_{AUXM}|$
2. Common mode voltage is equal to V_{bias_Tx} (or V_{bias_Rx}) voltage.
3. Steady state common mode voltage shift between transmit and receive modes of operation.
4. Total drive current of the transmitter when it is shorted to its ground.
5. All DisplayPort Main Link lanes as well as AUX CH must be AC coupled. AC coupling capacitors must be placed on the transmitter side. Placement of AC coupling capacitors on the receiver side is optional.

20.6.1.6 DDC Signal DC Specification

Table 126. DDC Signal DC Specification (DCC_DATA, DDC_CLK)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	MIPI_V1P8_S4			V	
V_{IH}	Input High Voltage	0.65*V_{REF}			V	1
V_{IL}	Input Low Voltage			0.35*V_{REF}	V	2
V_{OL}	Output Low Voltage			0.4	V	3
I_i	Input Pin Leakage	-30		30	µA	4

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.
3. 3mA sink current.
4. For VIN between 0V and CORE_VCC_S0iX. Measured when driver is tri-stated.

Table 127. DDC Misc Signal DC Specification (HPD, BKLTCTL, VDDEN, BKLTEN)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	MIPI_V1P8_S4			V	
V_{IH}	Input High Voltage	0.65*V_{REF}			V	1
V_{IL}	Input Low Voltage	0		0.35*V_{REF}	V	2
Z_{pu}	Pull up Impedance	40	50	60	Ω	3
Z_{pd}	Pull down Impedance	40	50	60	Ω	3
I_i	Input Pin Leakage	-20		20	µA	4

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value

2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.
3. Measured at CORE_VCC0_S0iX and CORE_VCC1_S0iX.
4. For VIN between 0V and CORE_VCC0_S0iX and CORE_VCC1_S0iX. Measured when driver is tri-stated.

Figure 35. Definition of Differential Voltage and Differential Voltage Peak-to-Peak

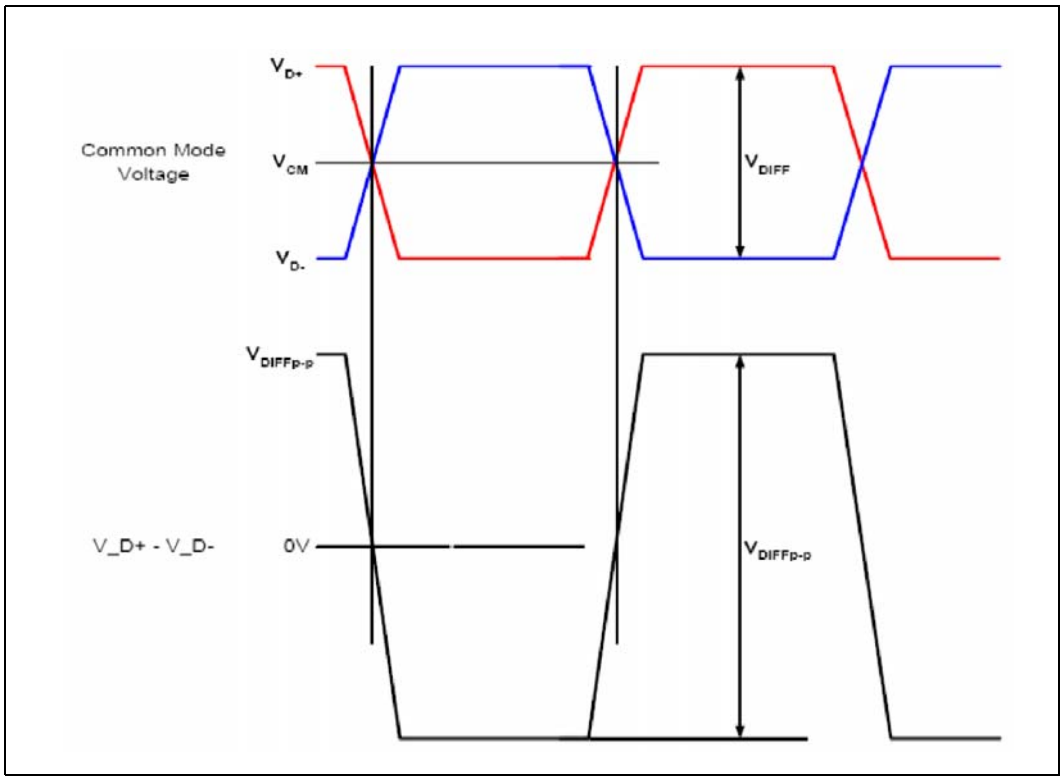
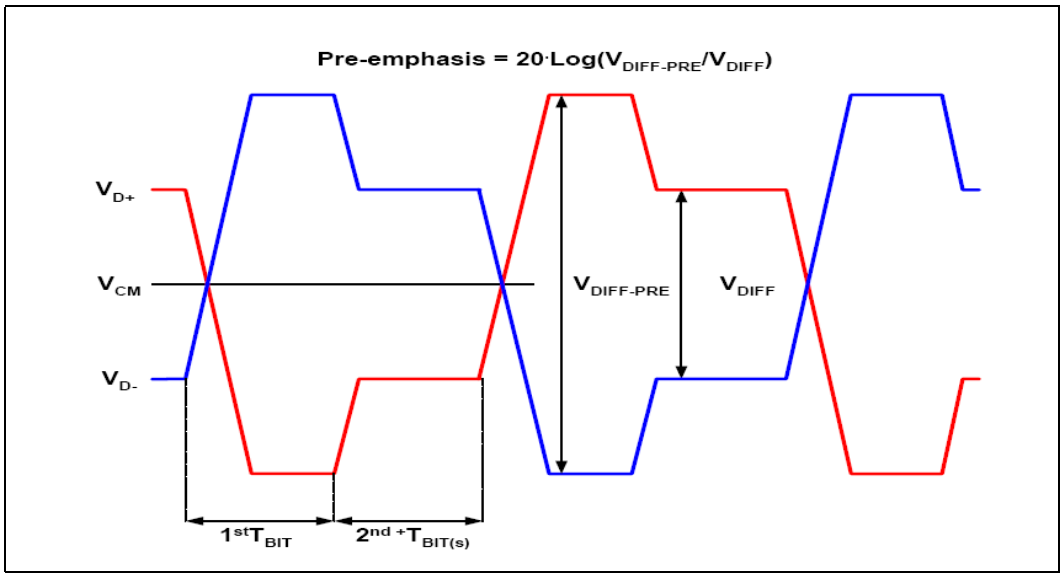
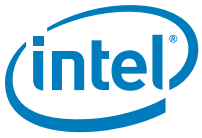


Figure 36. Definition of Pre-emphasis





20.6.1.7 MIPI DSI DC Specification

Table 128. MIPI DSI DC Specification

Symbol	Parameter	Min.	Nom.	Max.	Unit	Notes
ILEAK	Pin Leakage current	-10	–	10	µA	
MIPI DSI HS-TX Mode						
V_{CMTX}	HS transmit static common-mode voltage	150	200	250	mV	
$\|V_{CMTX(1,0)}\|$	V_{CMTX} mismatch when output is differential-1 or differential-0	–	–	5	mV	
$\|V_{OD}\|$	HS transmit differential voltage	140	200	270	mV	
$\|\Delta V_{OD}\|$	V_{OD} mismatch when output is Differential-1 or Differential-0	–	–	14	mV	
V_{OHHS}	HS output high voltage	–	–	360	mV	
Z_{OS}	Single-ended output impedance	40	50	62.5	Ω	
ΔZ_{OS}	Single-ended output impedance mismatch	–	–	10	%	
MIPI DSI LP-TX Mode						
V_{OH}	Thevenin output high level	1.1	1.2	1.3	V	
V_{OL}	Thevenin output low level	-50	–	50	mV	
Z_{OLP}	Output impedance of LP transmitter	50	–	–	Ω	1
MIPI DSI LP-RX Mode						
V_{IH}	Logic 1 input voltage	880	–	–	mV	
V_{IL}	Logic 0 input voltage, not in ULP state	–	–	550	mV	
V_{HYST}	Input hysteresis	25	–	–	mV	
V_{IHCD}	Logic 1 Contention threshold	450	–	–	mV	
V_{ILCD}	Logic 0 Contention threshold	–	–	200	mV	

NOTE: Deviates from MIPI D-PHY specification Rev 1.0, which has minimum ZOLP of 110 Ω.

20.6.2 MIPI-Camera Serial Interface (CSI) DC Specification

Table 129. MIPI HS-RX/MIPI LP-RX Minimum, Nominal, and Maximum Voltage Parameters

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
I_{LEAK}	Pin Leakage current	-10	–	10	µA	
MIPI-CSI HS-RX Mode						
$V_{CMRX(DC)}$	Common-mode voltage HS receive mode	70	–	330	mV	
V_{IDTH}	Differential input high threshold	–	–	70	mV	
V_{IDTL}	Differential input low threshold	-70	–	–	mV	
V_{IHHS}	Single-ended input high voltage	–	–	460	mV	
V_{ILHS}	Single-ended input low voltage	-40	–	–	mV	
$V_{TERM-EN}$	Single-ended threshold for HS termination enable	–	–	450	mV	
Z_{ID}	Differential input impedance	80	100	125	Ω	
MIPI-CSI LP-RX Mode						
V_{IH}	Logic 1 input voltage	880	–	–	mV	
V_{IL}	Logic 0 input voltage, not in ULP state	–	–	550	mV	
$V_{IL-ULPS}$	Logic 0 input voltage, ULP state	–	–	300	mV	
V_{HYST}	Input hysteresis	25	–	–	mV	

20.6.3 SDIO DC Specification

Table 130 provides the SDIO DC Specification, for all other DC Specifications not listed in Table 130, refer to Table 147, "GPIO 1.8V Core Well Signal Group DC Specification.

Table 130. SDIO DC Specification

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
V_{OH}	Output High Voltage	1.4	–	–	V	Measured at I_{OH} maximum.
I_{OH}/I_{OL}	Current at VoL/Voh	-2	–	–	mA	

20.6.4 SD Card DC Specification

Table 131 provides the SD Card DC Specification, for all other DC Specifications not listed in Table 131, refer to Table 147, "GPIO 1.8V Core Well Signal Group DC Specification.

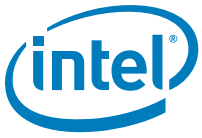


Table 131. SD Card DC Specification

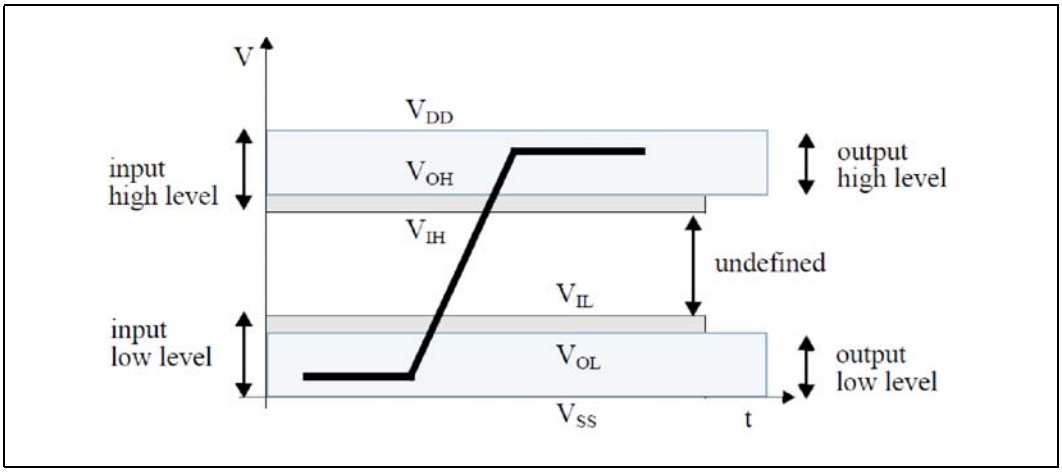
Symbol	Parameter	Min.	Max.	Unit
V_{REF}	I/O Voltage	SDIO_V3P3A_V1P8A_G3		
$V_{OH(3.3)}$	Output High Voltage	0.75*V_{REF}	–	V
$V_{OL(3.3)}$	Output Low Voltage	-	0.125*V_{REF}	V
$V_{IH\ (3.3)}$	Input High Voltage (3.3 V)	0.625*V_{REF}	V_{REF}+0.3	V
$V_{IL\ (3.3)}$	Input Low Voltage (3.3 V)	VSS-0.3	0.25*V_{REF}	V
$V_{OH(1.8)}$	Output High Voltage	1.40	–	V
$V_{OL(1.8)}$	Output Low Voltage	-	0.45	V
$V_{IH\ (1.8)}$	Input High Voltage (1.8 V)	1.27	2.00	V
$V_{IL\ (1.8)}$	Input Low Voltage (1.8 V)	VSS-0.3	0.58	V
I_{OH}/I_{OL}	Current at VoL/Voh	-2	2	mA
C_{LOAD}	total Load Capacitance	-	40	pF

20.6.5 eMMC 4.51 DC Specification

Table 132. eMMC 4.51 DC Electrical Specifications

Symbol	Parameter	Min	Max	Units
V_{REF}	I/O Voltage	GPIO_V1P8A_G3		
V_{OH}	Output HIGH voltage	V_{REF} - 0.45	-	V
V_{OI}	Output LOW voltage	-	0.45	V
V_{IH}	Input HIGH voltage	0.65 * V_{REF}	V_{REF} + 0.3	V
V_{IL}	Input LOW voltage	-0.3	0.35 * V_{REF}	V
C_L	Bus Signal Line capacitance	-	30	pF
I_{IL}	Input Leakage Current	-2	2	µA
I_{0L}	Output Leakage Current	-2	2	µA

Figure 37. eMMC 4.51 DC Bus Signal Level



20.6.6 JTAG DC Specification

Table 133. JTAG Signal Group DC Specification (JTAG_TCK, JTAG_TMS, JTAG_TDI, JTAG_TRST_N) (Sheet 1 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3				
V_{IH}	Input High Voltage	0.75*V_{REF}			V	1
V_{IL}	Input Low Voltage			0.35*V_{REF}	V	2
R_{wpu}	Weak Pull Up Impedance	2.5	5	7.5	kΩ	3

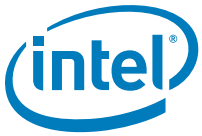


Table 133. JTAG Signal Group DC Specification (JTAG_TCK, JTAG_TMS, JTAG_TDI, JTAG_TRST_N) (Sheet 2 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
R_{wpd}	Weak Pull Down Impedance	2.5	5	7.5	kΩ	3
$R_{wpu-20K}$	Weak Pull Up Impedance 20K	12		28	kΩ	4
$R_{wpd-40K}$	Weak Pull Down Impedance 40K	20		70	kΩ	4

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.
3. Measured at GPIO_V1P8A_G3.
4. Rwpu_40k and Rwpd_40k are only used for JTAG_TRST#

Table 134. JTAG Signal Group DC Specification (JTAG_TDO)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3				
V_{IH}	Input High Voltage	$0.75*V_{REF}$			V	1
V_{IL}	Input Low Voltage			$0.45*V_{REF}$	V	2
Z_{pd}	Pull down Impedance	17.5		35	Ω	3
R_{wpu}	Weak Pull Up Impedance	-	-	-	kΩ	3
R_{wpd}	Weak Pull Down Impedance	-	-	-	kΩ	3

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.
3. Measured at GPIO_V1P8A_G3.

Table 135. JTAG Signal Group DC Specification (JTAG_PRDY#, JTAG_PREQ#)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3				
V_{IH}	Input High Voltage	$0.75*V_{REF}$			V	1
V_{IL}	Input Low Voltage			$0.45*V_{REF}$	V	2
Z_{pd}	Pull down Impedance	17.5		35	Ω	3
R_{wpu}	Weak Pull Up Impedance	-		-	kΩ	3

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.
3. Measured at GPIO_V1P8A_G3.

20.6.7 DDR3L-RS Memory Controller DC Specification

Table 136. DDR3L-RS Signal Group DC Specifications

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{IL}	Input Low Voltage			DDR_VREF - 200mV	V	1
V_{IH}	Input High Voltage	DDR_VREF + 200mV			V	2, 3
V_{OL}	Output Low Voltage		(DDR_VDDQG_S4 / 2)* (RON / (RON+RVTT_TERM))			3,4
V_{OH}	Output High Voltage		DDR_VDDQG_S4 - ((DDR_VDDQG_S4 /2)* (RON/(RON+RVTT_TERM))		V	3,4
I_{IL}	Input Leakage Current			5	µA	For all DDR Signals
R_{ON}	DDR3L-RS Clock Buffer strength	26		40	Ω	5
C_{IO}	DQ/DQS/DQS# DDR3L-RS IO Pin Capacitance		3.0		pF	

NOTES:
1. V_{IL} is defined as the maximum voltage level at the receiving agent that will be received as a logical low value. DDR_VREF is normally DDR_VDDQG_S4
2. V_{IH} is defined as the minimum voltage level at the receiving agent that will be received as a logical high value. DDR_VREF is normally DDR_VDDQG_S4
3. V_{IH} and V_{OH} may experience excursions above DDR_VDDQG_S4. However, input signal drivers must comply with the signal quality specifications.
4. RON is DDR driver resistance whereas RTT_TERM is DDR ODT resistance which is controlled by DDR.
5. DDR3L-1333 CLK buffer Ron is 26ohm and SR target is 4V/ns; DQ-DQS buffer Ron is 30ohms and SR target is 4V/ns; CMD/CTL buffer Ron is 20ohms and SR target is 1.8V/ns.

20.6.8 LPDDR3 Memory Controller DC Specification

Table 137. LPDDR3 Signal Group DC Specifications (Sheet 1 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
DDR_VDDQG_S4	I/O Supply Voltage	1.14	1.24	1.26	V	
V_{IL}	Input Low Voltage			DDR_VREF - 200 mV	V	
V_{IH}	Input High Voltage	DDR_VREF + 200 mV			V	
V_{OL}	Output Low Voltage	-	0.260	-	V	1,2
V_{OH}	Output High Voltage	-	0.960	-	V	1,2

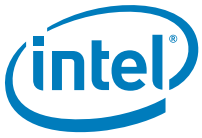


Table 137. LPDDR3 Signal Group DC Specifications (Sheet 2 of 2)

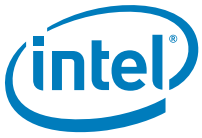
Symbol	Parameter	Min	Typ	Max	Units	Notes
I_{IL}	Input Leakage Current	-	5	-	µA	3,4
R_{ON}	Clock Buffer strength	26		40	Ω	
C_{IO}	IO Pin Capacitance		3.0		pF	

NOTES:
1. Vol & Voh is determined with 40ohm buffer strength setting into a 60ohm to 0.5x V1p5_ddr test load.
2. LPDDR3-1066 CLK buffer Ron is 35ohm and SR target is 2.5V/ns; DQ-DQS buffer Ron is 40ohms and SR target is 2V/ns; CMD/CTL buffer Ron is 30ohms and SR target is 1.5V/ns.
3. Applies to the pin to VCC or VSS leakage current.
4. Applies to the pin to pin leakage current.

20.6.9 USB 2.0 Host DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
Supply Voltage:						
VBUS	High-power Port	4.75		5.25	V	2
VBUS	Low-power Port	4.20		5.25		
Supply Current:						
ICCPRT	High-power Hub Port (out)	500			mA	
ICCUPT	Low-power Hub Port (out)	100			mA	
ICCHPF	High-power Function (in)			500	mA	
ICCLPF	Low-power Function (in)			100	mA	
ICCINIT	Unconfigured Function/Hub (in)			100	mA	
ICCSH	Suspended High-power Device			2.5	mA	15
ICCSL	Suspended Low-power Device			500	µA	
Input Levels for Low-/full-speed:						
VIH	High (driven)	2.0			V	4
VIHZ	High (floating)	2.7		3.6	V	4
VIL	Low			0.8	V	4
VDI	Differential Input Sensitivity	0.2			V	|(D+)-(D-)|;Figure; Note 4
VCM	Differential Common Mode Range	0.8		2.5	V	Includes VDI range; Figure; Note 4
Input Levels for High-speed:						

Symbol	Parameter	Min	Typ	Max	Units	Notes
VHSSQ	High-speed squelch detection threshold (differential signal amplitude)	100		150	mV	
VHSDSC	High speed disconnect detection threshold (differential signal amplitude)	525		625	mV	
	High-speed differential input signaling levels					16
VHSCM	High-speed data signaling common mode voltage range (guideline for receiver)	-50		500	mV	
Output Levels for Low-/full-speed:						
VOL	Low	0.0		0.8	V	4,5
VOH	High (Driven)	2.8		3.6	V	4,6
VOSE1	SE1	0.8			V	
VCRS	Output Signal Crossover Voltage	1.3		2.0	V	10
Output Levels for High-speed:						
VHSOI	High-speed idle level	-10		10	mV	
VHSOH	High-speed data signaling high	360		440	mV	
VHSOL	High-speed data signaling low	-10		10	mV	
VCHIRPJ	Chirp J level (differential voltage)	700		1100	mV	
VCHIRPK	Chirp K level (differential voltage)	-900		-500	mV	
Decoupling Capacitance:						
CHPB	Downstream Facing Port Bypass Capacitance (per hub)	120			µF	
CRPB	Upstream Facing Port Bypass Capacitance	1.0		10.0	µF	9
Input Capacitance for Low-/full-speed:						
CIND	Downstream Facing Port			150	pF	2
CINUB	Upstream Facing Port (w/o cable)			100	pF	3
CEDGE	Transceiver edge rate control capacitance			75	pF	
Input Impedance for High-speed:						
	TDR spec for high-speed termination					
Terminations:						
RPU	Bus Pull-up Resistor on Upstream Facing Port	1.425		1.575	kΩ	1.5 kΩ ±5%



Symbol	Parameter	Min	Typ	Max	Units	Notes
RPD	Bus Pull-down Resistor on Downstream Facing Port	14.25		15.75	kΩ	1.5 kΩ ±5%
ZINP	Input impedance exclusive of pull-up/pull-down (for low-/full speed)	300			kΩ	
VTERM	Termination voltage for upstream facing port pull-up (RPU)	3.0		3.6	V	
Terminations in High-speed:						
VHSTERM	Termination voltage in high speed	-10		10	mV	
RTERM	High Speed Termination	40	45	50	Ω	
VBUSD	VBUS Voltage drop for detachable cables	-	-	1	mV	

NOTES:

1. Measured at A plug.
2. Measured at A receptacle.
3. Measured at B receptacle.
4. Measured at A or B connector.
5. Measured with RL of 1.425 kΩ to 3.6 V.
6. Measured with RL of 14.25 kΩ to GND.
7. Timing difference between the differential data signals.
8. Measured at crossover point of differential data signals.
9. The maximum load specification is the maximum effective capacitive load allowed that meets the target VBUS drop of 330 mV.
10. Excluding the first transition from the Idle state.
11. The two transitions should be a (nominal) bit time apart.
12. For both transitions of differential signaling.
13. Must accept as valid EOP.
14. Single-ended capacitance of D+ or D- is the capacitance of D+/D- to all other conductors and, if present, shield in the cable. That is, to measure the single-ended capacitance of D+, short D-, VBUS, GND, and the shield line together and measure the capacitance of D+ to the other conductors .
15. For high power devices (non-hubs) when enabled for remote wakeup.
16. Specified by eye pattern templates

20.6.10 USB 3.0 DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
UI	Unit Interval	199.94		200.06	ps	1
$V_{TX-DIFF-PP}$	Differential peak-peak Tx voltage swing	0.9	1	1.05	V	
$V_{TX-DIFF-PP-LOW}$	Low-Power Differential peak-peak Tx voltage swing	0.4		1.2	V	2
$V_{TX-DE-RATIO}$	Tx De-Emphasis	3.45	3.5	3.65	dB	

Symbol	Parameter	Min	Typ	Max	Units	Notes
$R_{TX-DIFF-DC}$	DC differential impedance	88		92	Ω	
$V_{TX-RCV-DETECT}$	The amount of voltage change allowed during Receiver Detection			0.6	V	3
$C_{AC-COUPLING}$	AC Coupling Capacitor	75		200	nF	4
$t_{CDR\_SLEW\_MAX}$	Maximum slew rate			10	ms/s	

NOTES:
1. The specified UI is equivalent to a tolerance of 300ppm for each device. Period does not account for SSC induced variations.
2. There is no de-emphasis requirement in this mode. De-emphasis is implementation specific for this mode.
3. Detect voltage transition should be an increase in voltage on the pin looking at the detect signal to avoid a high impedance requirement when an "off" receiver's input goes below output.
4. All transmitters shall be AC coupled. The AC coupling is required either within the media or within the transmitting component itself.

20.6.11 LPC DC Specification

Table 138. LPC 1.8V Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{IH}	Input High Voltage	1.5	1.8	1.8 +0.5	V	
V_{IL}	Input Low Voltage	-0.5	0	0.8	V	
V_{OH}	Output High Voltage	0.9 x 1.8			V	
V_{OL}	Output Low Voltage			0.1 x 1.8	V	
I_{OH}	Output High Current		0.5		mA	
I_{OL}	Output Low Current		-1.5		mA	
I_{LEAK}	Input Leakage Current	-10		10	µA	
C_{IN}	Input Capacitance			10	pF	

Table 139. LPC 3.3V Signal Group DC Specification (Sheet 1 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{IH}	Input High Voltage	2.0	3.3	3.3 +0.5	V	1
V_{IL}	Input Low Voltage	-0.5	0	0.8	V	2
V_{OH}	Output High Voltage	2.5			V	3
V_{OL}	Output Low Voltage			0.4	V	3
I_{OH}	Output High Current		0.5		mA	3

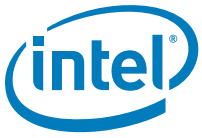


Table 139. LPC 3.3V Signal Group DC Specification (Sheet 2 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
I_{OL}	Output Low Current		-1.5		mA	3
I_{LEAK}	Input Leakage Current	-10		10	µA	
C_{IN}	Input Capacitance			10	pF	

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value, Applies to LPC_AD[3:0], LPC_CLKRUN_N
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value. Applies to LPC_AD[3:0], ILB_LPC_CLKRUN_N
3. V_{OH} is tested with Iout=500uA, V_{OL} is tested with Iout=1500uA
4. Applies to LPC_AD[3:0],LPC_CLKRUN_N and LPC_FRAME_N
5. LPC_SERIRQ is always a 1.8V I/O irrespective of the value of LPC_V1P8V3P3_S4.

20.6.12 SPI DC Specification

Table 140. SPI Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_1P8A_G3			V	3
V_{IH}	Input High Voltage	0.65 * V_{REF}			V	2
V_{IL}	Input Low Voltage	-0.5		0.35 * V_{REF}	V	2
V_{OH}	Output High Voltage	V_{REF} - 0.45		1.8V	V	1
V_{OL}	Output Low Voltage			0.45	V	1
I_{OH}	Output High Current			2	mA	1
I_{OL}	Output Low Current	-2			mA	1

NOTES:
1. Applies to SPI1_CS[1:0], SPI1_CLK, SPI1_MOSI
2. Applies to SPI1_MISO and SPI1_MOSI
3. The I/O buffer supply voltage is measured at the SoC package pins. The tolerances shown are inclusive of all noise from DC up to 20 MHz. In testing, the voltage rails should be measured with a bandwidth limited oscilloscope that has a rolloff of 3 dB/decade above 20 MHz.

20.6.13 Power Management/Thermal (PMC) & RTC DC Specification

Table 141. Power Management 1.8V Suspend Well Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_1P8A_G3			V	
V_{IH}	Input High Voltage	0.65 * V_{REF}			V	2
V_{IL}	Input Low Voltage	-0.5		0.35 * V_{REF}	V	2
V_{OH}	Output High Voltage	V_{REF} - 0.45		1.8V	V	1

Table 141. Power Management 1.8V Suspend Well Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{OL}	Output Low Voltage			0.45	V	1
I_{OH}	Output High Current			2	mA	1
I_{OL}	Output Low Current	-2			mA	1

NOTES:
1. The data in this table apply to signals - PMC_ACPRESENT, PMC_BATLOW_N, PMC_PLTRST_N, PMC_PWRBTN_N,PMC_SLP_S4_N, PMC_SUS_STAT_N, PMC_SUSCLK[3:0], PMC_SUSPWRDNACK
2. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
3. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.

Table 142. PMC_RSTBTN# 1.8V Core Well Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
VREF	I/O Voltage	UNCORE_V1P8_G3			V	
V_{IH}	Input High Voltage	0.65* VREF			V	1
V_{IL}	Input Low Voltage			0.35* VREF	V	2

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.

Table 143. Power Management & RTC Well Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
VREF	I/O Voltage	RTC_V3P3RTC_G5				
V_{IH}	Input High Voltage	2.0	-	-	V	1
V_{IL}	Input Low Voltage	-	-	0.78	V	2

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.

Table 144. RTC Well DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{IH}	Input High Voltage	2.3	-	-	V	1
V_{IL}	Input Low Voltage	-	-	0.78	V	1

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.

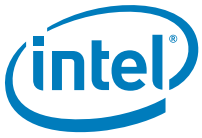


Table 145. PROCHOT# Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	1p8V				
V_{IH}	Input High Voltage	0.75*V_{REF}		V_{REF}	V	1
V_{IL}	Input Low Voltage			0.45*V_{REF}	V	2
V_{OL}	Output Low Voltage			0.35 * V_{REF}	V	
I_{OL}	Output Low Current			-5	mA	

NOTES:
1. V_{IH} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical high value
2. V_{IL} is defined as the minimum voltage level at a receiving agent that will be interpreted as a logical low value.

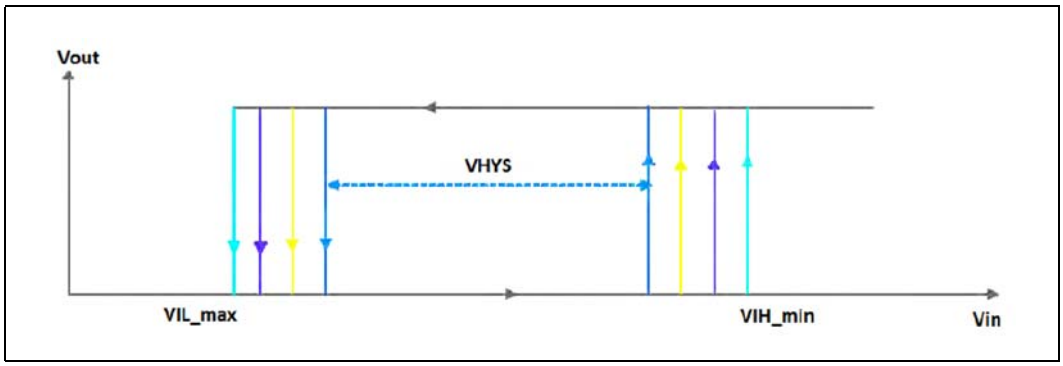
20.6.14 SVID DC Specification

Table 146. SVID Signal Group DC Specification (SVID_DATA, SVID_CLK, SVID_ALERT_N)

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3				
V_{IH}	Input High Voltage	0.65*V_{REF}			V	1
V_{IL}	Input Low Voltage			0.35*V_{REF}	V	1
V_{OH}	Output High Voltage	V_{REF} - 0.45		V_{REF}	V	1
V_{OL}	Output Low Voltage			0.45	V	4
V_{HYS}	Hysteresis Voltage	0.1			V	
R_{ON}	BUffer on Resistance	40		60	Ω	2
I_L	Leakage Current	-10		10	µA	3
C_{PAD}	Pad Capacitance			9	pF	4
V_{PIN}	Pin Capacitance			10	pF	
Z_{pd}	Pull down Impedance	35	50	70	Ω	

NOTES:
1. GPIO_V1P8A_G3 refers to instantaneous voltage VSS_SENSE
2. Measured at 0.31 * GPIO_V1P8A_G3
3. V_{IN} between 0V and GPIO_V1P8A_G3
4. CPAD includes die capacitance only. No package parasitic included.

Figure 38. Definition of VHYS in Table 169



20.6.15 GPIO DC Specification

GPIO Buffer is used across various interfaces on the SoC such as, GPIOs, I2C, I2S, SPI, SDIO, SVID, UART, JTAG and ULPI.

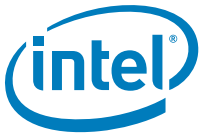
Table 147. GPIO 1.8V Core Well Signal Group DC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3				
V_{IH}	Input High Voltage	0.65*V_{REF}			V	
V_{IL}	Input Low Voltage			0.35 * V_{REF}	V	
V_{OH}	Output High Voltage	V_{REF} - 0.45		V_{REF}	V	
V_{OL}	Output Low Voltage			0.45	V	
V_{Hys}	Input Hysteresis	0.1			V	
I_L	Leakage Current	-2		2	mA	
C_{LOAD}	Load Capacitance	2		75	pF	

20.6.16 SIO - I2C DC Specification

Table 148. I2C Signal Electrical Specifications

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{REF}	I/O Voltage	GPIO_V1P8A_G3			V	
V_{IH}	Input High Voltage	0.7 * V_{REF}			V	
V_{IL}	Input Low Voltage			0.3 * V_{REF}	V	
V_{OL}	Output Low Voltage			0.2 * V_{REF}	V	
V_{Hys}	Input Hysteresis	0.1			V	
C_{PIN}	Pin Capacitance	2		5	pF	



20.6.17 SIO - UART DC Specification

Refer to the GPIO Buffer (1.8V) DC Specification, mentioned Section 20.6.15, "GPIO DC Specification" on page 290

20.6.18 I²S (Audio) DC Specification

Refer to the GPIO Buffer (1.8V) DC Specification, mentioned Section 20.6.15, "GPIO DC Specification" on page 290

20.6.19 PCI Express DC Specification

Table 149. PCI Express DC Receive Signal Characteristics

Symbol	Parameter	Min	Typ	Max	Unit	Notes
$V_{RXDIFF\ Gen1}$	Differential RX Peak to Peak	175		1200	mV	1
$V_{RXDIFF\ Gen2}$	Differential RX Peak to Peak	100		1200	mV	1

NOTE:
1. PCI Express differential peak to peak = 2*|RXp[x] – RXn[x]|

Table 150. PCI Express DC Transmit Characteristics

Symbol	Parameter	Min	Typ	Max	Unit	Notes
V_{TXDIFF}	Differential TX Peak to Peak	800		1200	mV	1
$V_{TXDIFF-LP}$	Differential TX Peak to Peak (low power mode)	400		1200	mV	1

NOTE:
1. PCI Express differential peak to peak = 2*|TXp[x] – TXn[x]|

Table 151. PCI Express DC Clock Request Input Signal Characteristics

Symbol	Parameter	Min	Typ	Max	Unit	Notes
V_{REF}	I/O Voltage	UNCORE_V1P8_S4				
V_{IL}	Input Low Voltage			$0.3*V_{REF}$	V	1
V_{IH}	Input High Voltage	$0.65*V_{REF}$			V	1

NOTE:
1. 3.3 V refers to UNCORE_3P3_S0 for signals in the core well. See Chapter 2, "Physical Interfaces" for signal and power well association.

20.7 AC Specifications

The timings specified in this section are defined at the SoC pads. Therefore, proper simulation of the signals is the only means to verify proper timing and signal quality.

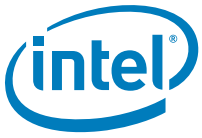
See Chapter 2, "Physical Interfaces" for signal definitions and Chapter 22, "Ballout and Ball Map" for the ball map. Generic timing diagrams can be found in "General AC Timing Diagrams".

The timings specified in this section should be used in conjunction with the SoC signal integrity models provided by Intel.

See the following AC Specifications in this section:

- "Platform Clocks AC Specification"
- "SVID AC Specification"
- "DDR3L-RS Memory Controller AC Specification"
- "LPDDR3 Memory Controller AC Specification"
- "Display AC Specifications"
- "MIPI-Camera Serial Interface (CSI) AC Specification"
- "SD Card AC Specification"
- "SDIO AC Specification"
- "eMMC 4.51 AC Specification"
- "USB 2.0 Host AC Specification"
- "USB 2.0 HSIC AC Specification"
- "I^2S (Audio) AC Specification"
- "PMC - Suspended Clock AC Specification"
- "SPI AC Specification"
- "PCU - LPC AC Specification"
- "I^2C AC Specification"
- "UART AC Specification"
- "JTAG AC Specification"
- "PCI Express AC Specification"
- "General AC Timing Diagrams"

Note: Care should be taken to read all notes associated with a particular timing parameter.



20.7.1 Platform Clocks AC Specification

Table 152. 19.2 MHz Platform Clock AC Specification

Symbol	Parameter	Min.	Typ	Max.	Unit	Notes
F_{PLT}	Frequency		19.2		MHz	
T_{DC}	Duty Cycle	45		55	%	
$T_{RISE/FALL}$	Minimum and Maximum Rise/Fall Time	5		20	ns	
V_{SWING}	Voltage Swing		1.8		V	
$T_{PEAKJIT}$	Peak Jitter (c-c)	-300		300	ps	
T_{PERJIT}	Period Jitter			550	ps	

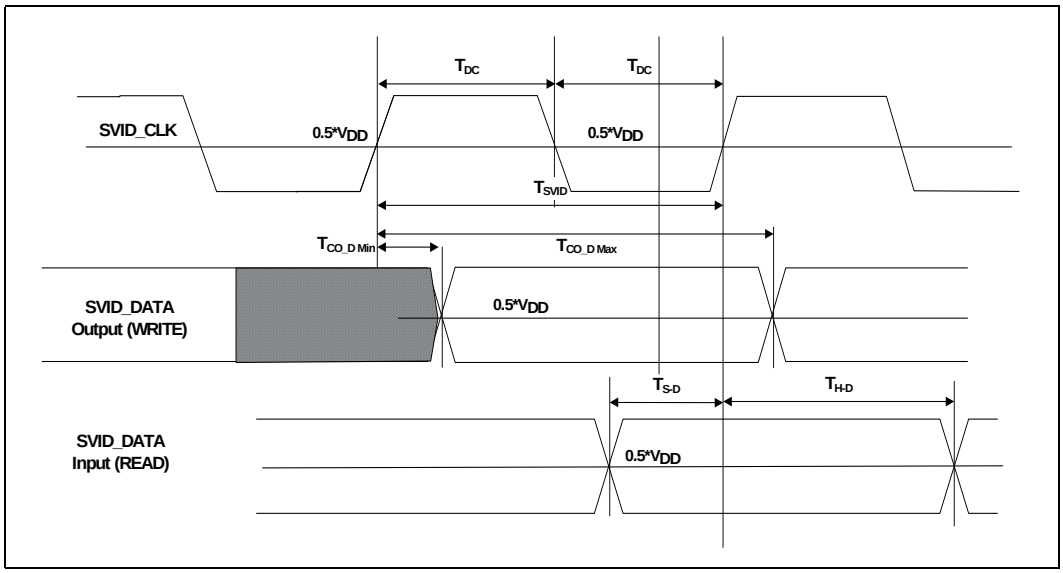
20.7.2 SVID AC Specification

Table 153. SVID AC Specification

Symbol	Parameter	Min.	Typ	Max.	Unit	Figure	Notes
F_{CLK}	SVID_CLK Frequency			25	MHz	39	1
T_{DUTY}	SVID_CLK Duty Cycle	45		55	%		1
T_{SU}	SVID_DATA Input Setup Time	7.5			ns	39	2,3,4
T_{HD}	SVID_DATA Input Hold Time	0			ns	39	2,3,4
T_{CO}	Rising Edge SVID_CLKC to SVID_DATA Output delay	-2.0		2.2	ns	39	1,4
TRISE/ FALL	Minimum and Maximum Rise/Fall Time	0		1	ns		5

NOTES:
1. Timing is measured from 50% to 50% Vdd Measured from 30–70
2. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
3. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
4. SoC output timings are measured at SoC pad with a test load of 12pF
5. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 35%/ 65%

Figure 39. SVID Timing Diagram



20.7.3 DDR3L-RS Memory Controller AC Specification

Note: The contents of this section are only valid for DDR_VDDQG_S4 = 1.35V

Table 154. DDR3L-RS Interface Timing Specification (Sheet 1 of 3)

Symbol	Parameter	Min	Max	Unit	Figure	Notes
DDR3L-RS Electrical Characteristic and AC timings at 1600 MT/s						
T_{SLR_D}	DQ, DQSP, DQSN Input Slew Rate	3	5.5	V/ns		
System Memory Clock Timings						
$T_{CK(AVG)}$	Average CK Period		1.875	ns		
T_{CH}	Average CK High Time	0.45		tCKAVG		
T_{CL}	Average CK Low Time	0.45		tCKAVG		
T_{SKEW}	Skew between any System Memory Differential Clock Pair (CKP/CKN)		30	ps		
System Memory Command Signal Timings						
T_{CMD} (tCMDVB+tCMDVA)	Total CMD Buffer window available for command buffers (RAS#, CAS#, WE#, BS[2:0], MA)	1380		ps		1
System Memory Control Signal Timings						

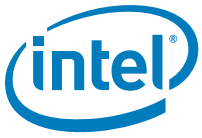


Table 154. DDR3L-RS Interface Timing Specification (Sheet 2 of 3)

Symbol	Parameter	Min	Max	Unit	Figure	Notes
T_{CTL} (tCTLVB + tCTLVA)	Total Control buffer Window available for Control buffers (CS#, CKE)	1400		ps		2
System Memory Data and Strobe Signal Timings						
$T_{DVB}+T_{VDA}$	Data, DQ and DM timing window available at the interface output for write commands. tDVB is data available before strobe and tDVA is data available after corresponding slope.	645		ps		3
$T_{SU} + T_{HD}$	Data, DQ Input Setup Plus Hold Time requirement for successful Read operation. These Setup and Hold numbers are measured w.r.t. corresponding strobe or Falling Edge	310		ps		4
T_{DQSS}	Strobe to rising clock edge during write.	-120		120	ps	
T_{WPRE}	DQSP/N Preamble duration (one dummy cycle)	0.9		tCKAVG		
T_{WPST}	DQSP/N Postamble Duration	0.4		tCKAVG		
DDR3L-RS Electrical Characteristic and AC timings at 1333 MT/s. DRAM_VDD_S4 = 1.35 V						
T_{SLR_D}	DQ, DQSP, DQSN Input Slew Rate	3	5.5	V/ns		
System Memory Clock Timings						
$T_{CK(AVG)}$	Average CK Period		1.5	ns		
T_{CH}	Average CK High Time	0.45		tCKAVG		
T_{CL}	Average CK Low Time	0.45		tCKAVG		
T_{SKEW}	Skew between any System Memory Differential Clock Pair (CKP/CKN)		30	ps		
System Memory Command Signal Timings						
T_{CMD} (tCMDVB+tCMDVA)	Total CMD Buffer window available for command buffers (RAS#, CAS#, WE#, BS[2:0], MA)	1075		ps		1
System Memory Control Signal Timings						
T_{CTL} (tCTLVB + tCTLVA)	Total Control buffer Window available for Control buffers (CS#, CKE)	1125		ps		2
System Memory Data and Strobe Signal Timings						

Table 154. DDR3L-RS Interface Timing Specification (Sheet 3 of 3)

Symbol	Parameter	Min	Max	Unit	Figure	Notes
$T_{DVB}+T_{VDA}$	Data, DQ and DM timing window available at the interface output for write commands. tDVB is data available before strobe and tDVA is data available after corresponding slope.	495		ps		3
$T_{SU} + T_{HD}$	Data, DQ Input Setup Plus Hold Time requirement for successful Read operation. These Setup and Hold numbers are measured w.r.t. corresponding strobe or Falling Edge	255		ps		4
T_{DQSS}		-120		120	ps	
T_{WPRE}	DQSP/N Preamble duration (one dummy cycle)	0.9		tCKAVG		
T_{WPST}	DQSP/N Postamble Duration	0.4		tCKAVG		

NOTES:
1. The CMD time is measured w.r.t. differential crossing of DRAM_CKP and DRAM_CKN. The tCMDVB and tCMDVA will be adjusted for proper CMD Setup and Hold time requirement at DRAM. The command timing assumes CMD-1N Mode.
2. The CTL time is measured w.r.t. differential crossing of DRAM_CKP and DRAM_CKN. The tCTLVB and tCTLVA will be adjusted for proper CTL Setup and Hold time requirement at DRAM.
3. The accurate strobe placement using write training algorithm will be performed which will guarantee the required Data setup/hold time w.r.t. strobe differential crossing at the DRAM input.
4. The Read training algorithm will center the DQS internally inside DRAM interface in order to have equal tSU and tHD timings.
5. All the timing windows are measured at 50% of the respective DRAM signal swing.

Figure 40. DDR3L-RS DQ Setup/Hold Relationship to/from DQSP/DQSN (Read Operation)

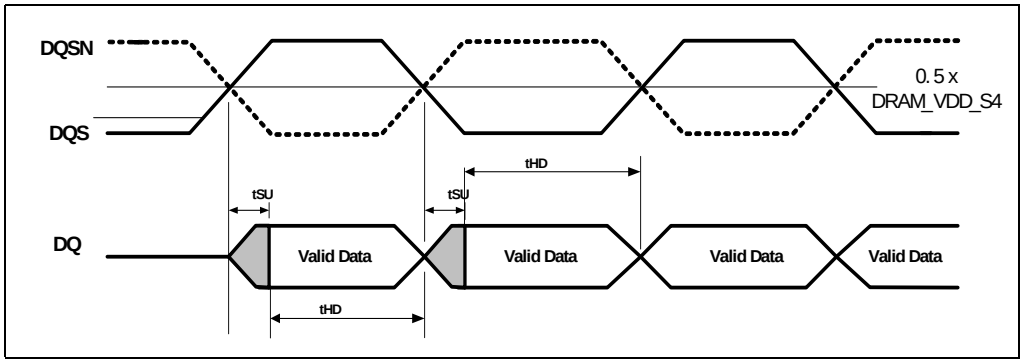


Figure 41. DDR3L-RS DQ and DM Valid before and after DQSP/DQSN (Write Operation)

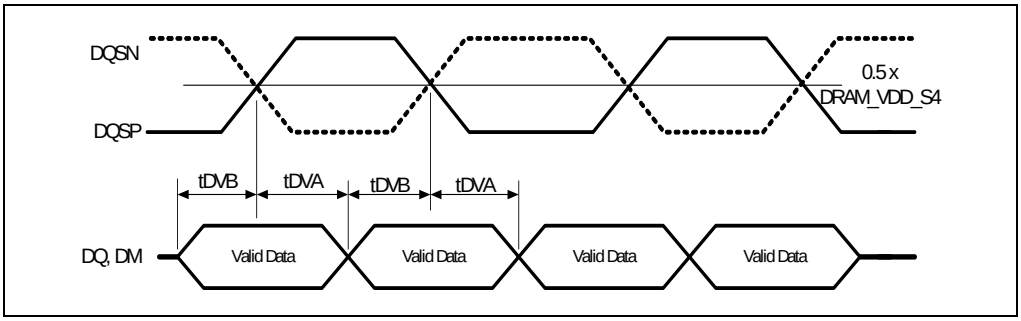


Figure 42. DDR3L-RS Write Pre-amble Duration

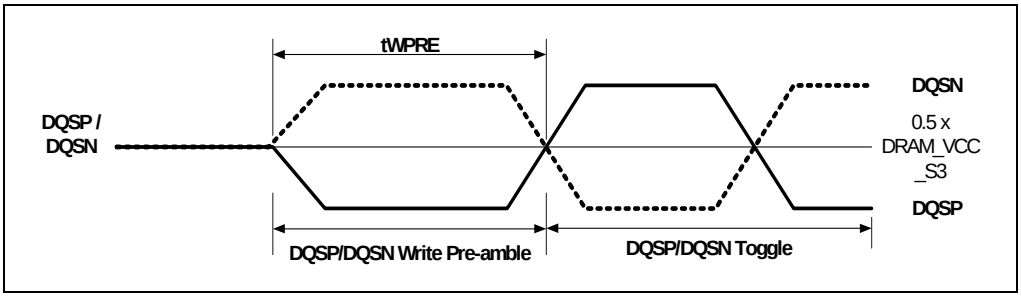


Figure 43. DDR3L-RS Write Post-amble Duration

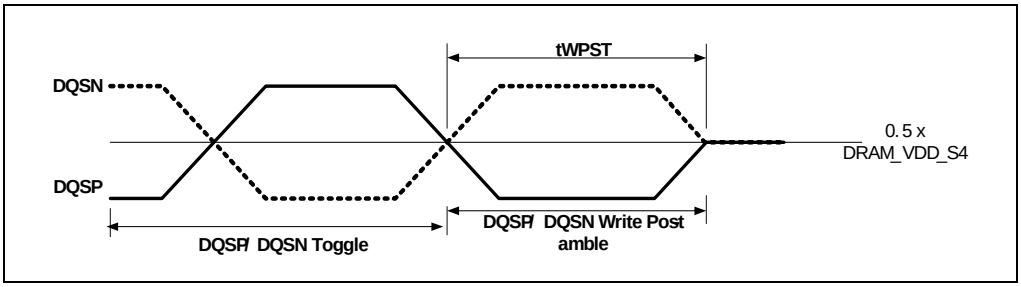


Figure 44. DDR3L-RS Command Signals Valid before and after CK Rising Edge

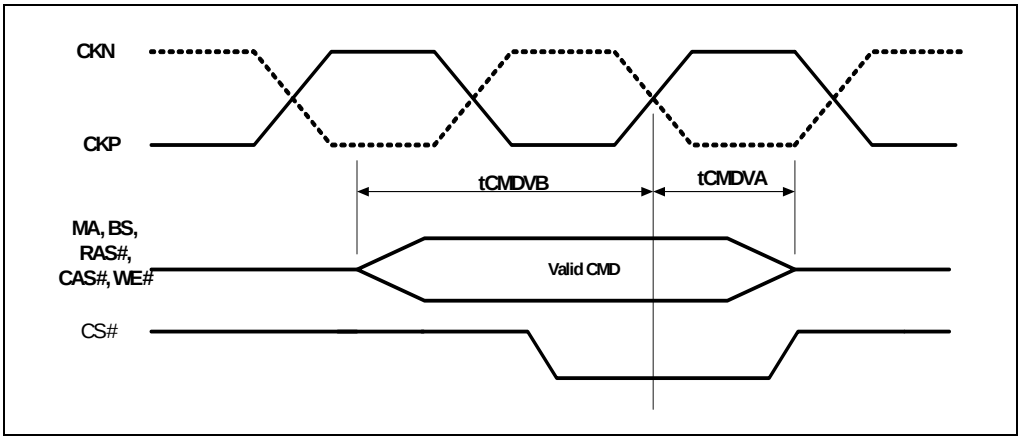


Figure 45. DDR3L-RS CKE Valid before and after CK Rising Edge

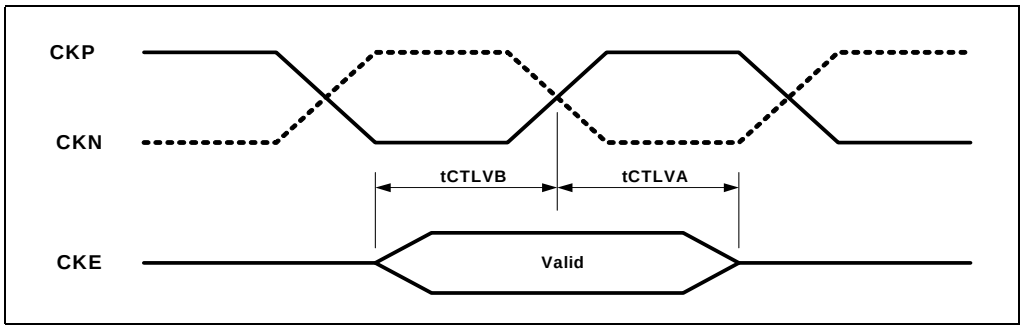


Figure 46. DDR3L-RS CS_N Valid before and after CK Rising Edge

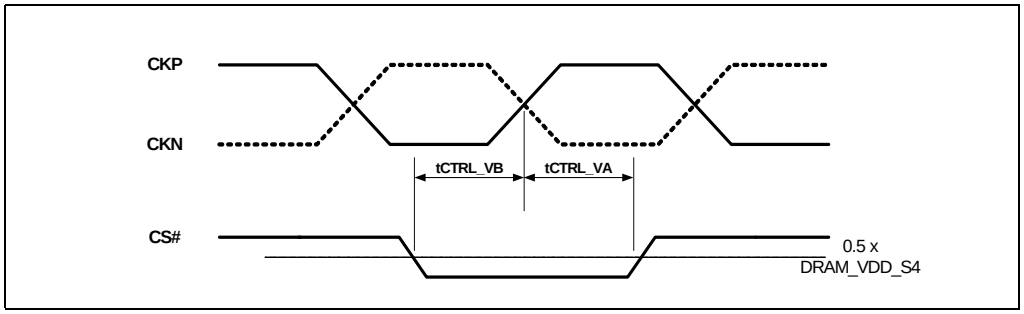


Figure 47. DDR3L-RS ODT Valid before CK Rising Edge

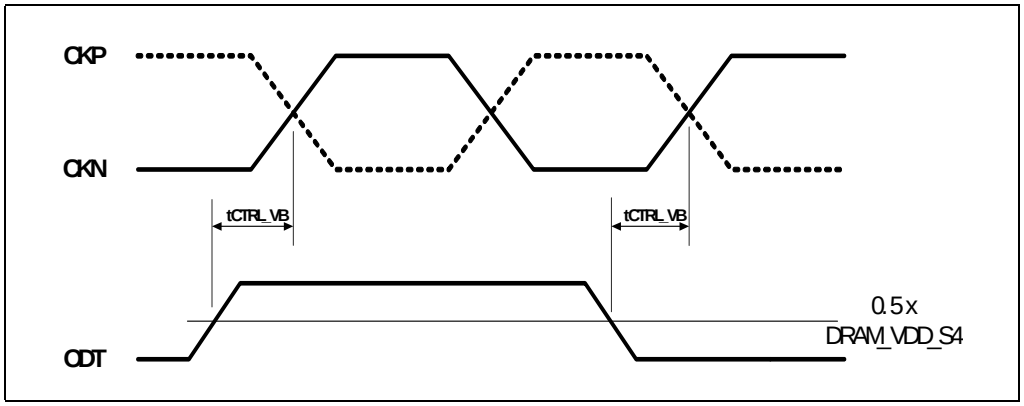


Figure 48. DDR3L-RS Clock Cycle Time

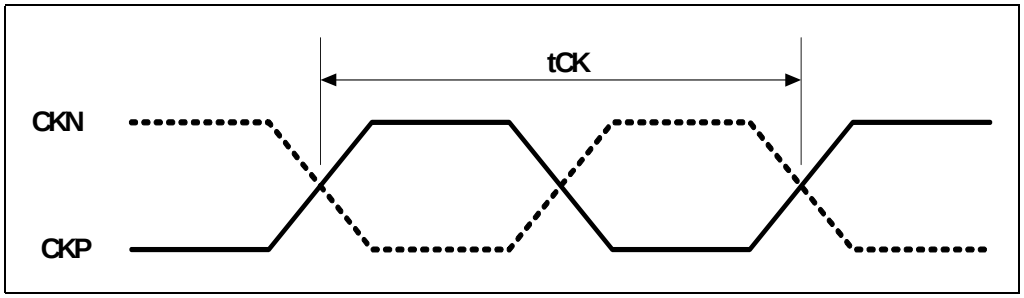
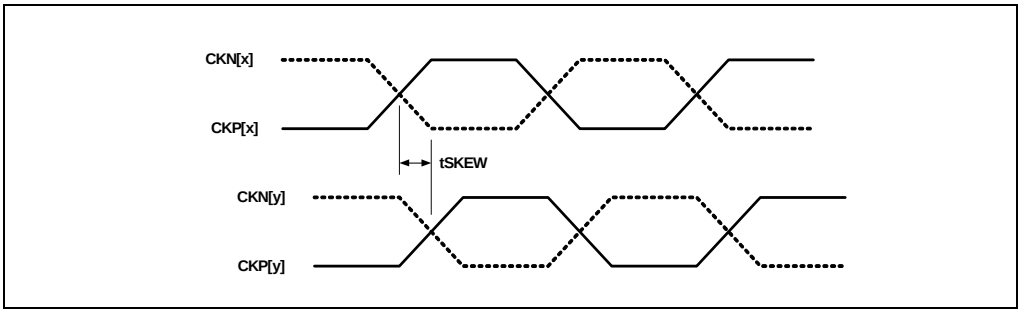


Figure 49. DDR3L-RS Skew between System Memory Differential Clock Pairs (CKP/CKN)



NOTE: x represents one differential clock pair, and y represents another differential clock pair within same channel.

Figure 50. DDR3L-RS CK High Time

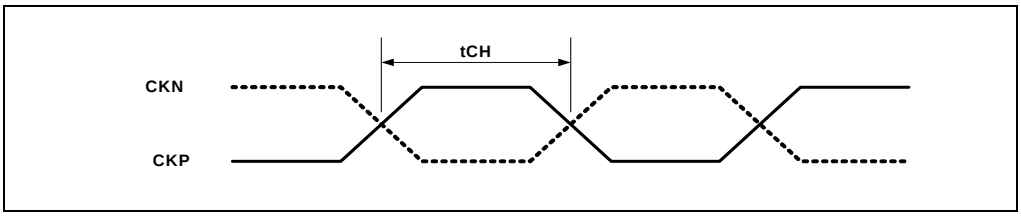


Figure 51. DDR3L-RS CK Low Time

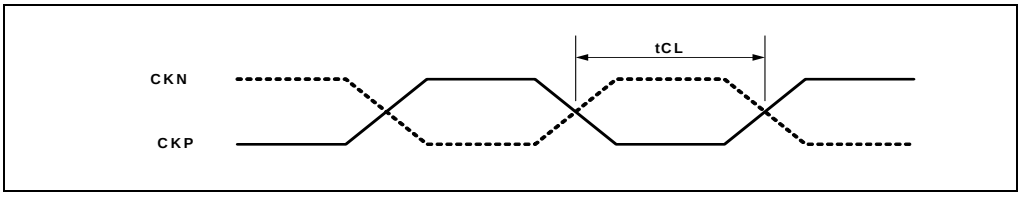


Figure 52. DDR3L-RS DQS Falling Edge Output Access Time to CK Rising Edge

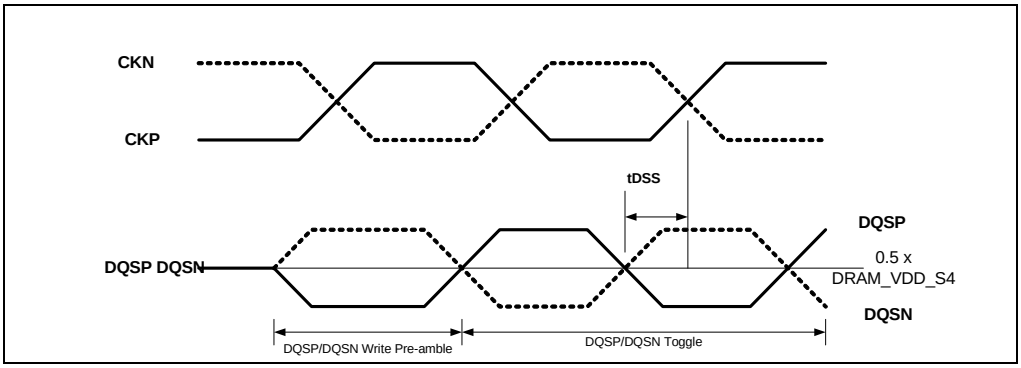


Figure 53. DDR3L-RS DQS Falling Edge Output Access Time From CK Rising Edge

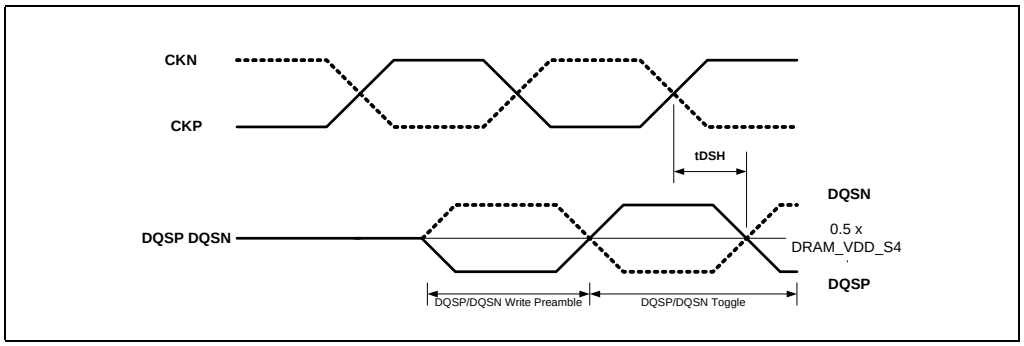
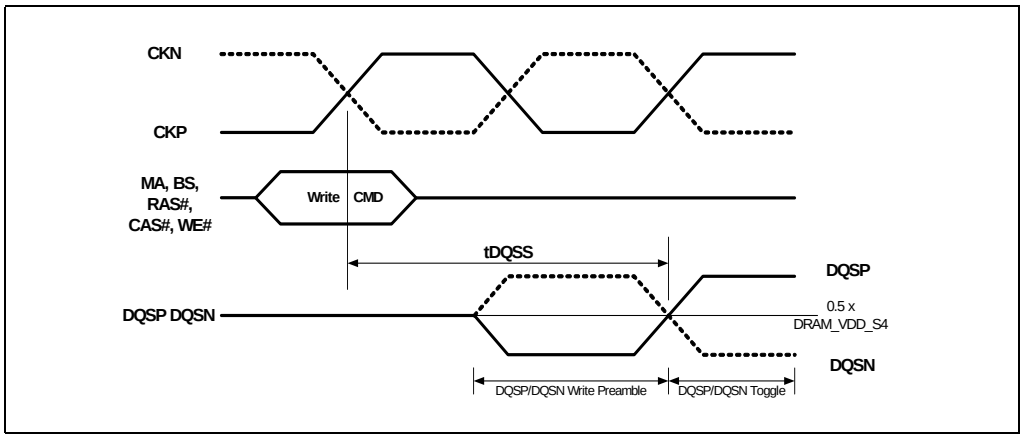


Figure 54. DDR3L-RS CK Rising Edge Output Access Time to the 1st DQS Rising Edge



20.7.4 LPDDR3 Memory Controller AC Specification

Note: The reference point for all LPDDR3 AC timing measurements is 0.5 * DDR_VDDQG_S4

Table 155. LPDDR3 Interface Timing Specification (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Figure	Notes
LPDDR3 Electrical Characteristic and AC timings at 1600 MT/s. DDR_VDDQG_S4 = 1.2 V ±0.1V						
T_{SLR_D}	DRAM_DQ[63:0], DRAM_DQSP[7:0], DRAM_DQSN[7:0] Input Slew Rate	1.5		V/ns		6
System Memory Clock Timings						
$T_{CK(AVG)}$	Average CK Period	1.876		ns		
T_{CH}	Average CK High Time	0.45	0.55	ns		
T_{CL}	Average CK Low Time	0.45	0.55	ns		
$T_{JIT(period)}$	Clock period Jitter	-76		ps		8,9
$T_{JIT(c2c)}$	Cycle to cycle Clock Period Jitter	-180		ps		8,9

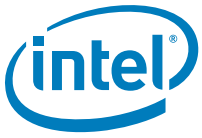


Table 155. LPDDR3 Interface Timing Specification (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Figure	Notes
T_{SKEW}	Skew between any System Memory Differential Clock Pair		30	ps		
System Memory Command Signal Timings						
T_{CMD} (T_{CMD_VB}+T_{CMD_VA})	Total CMD Buffer window available for command buffers	250		ps		2,4,10
System Memory Control Signal Timings						
T_{CTL} (T_{CTRL_VB} + T_{CTRL_VA})	Total Control buffer Window available for Control buffers	250		ps		4,10
System Memory Data and Strobe Signal Timings						
T_{DVB}+T_{VDA}	Data, DRAM_DQ[63:0] and DRAM_DM[7:0] timing window available at the interface output for write commands. tDVB is data available before strobe and tDVA is data available after corresponding slope.	250		ps		5,10
T_{SU} + T_{HD}	Data, DRAM_DQ[63:0] Input Setup Plus Hold Time requirement for successful Read operation. These Setup and Hold numbers are measured w.r.t. corresponding strobe or Falling Edge	220		ps		1,5,6, 7

NOTES:

1. Data to Strobe read setup and Data from Strobe read hold minimum requirements specified at the SoC pad are determined with the minimum Read DRAM_DQ/DRAM_DQ# Delay.
2. Command Timings are based off 1/2N Command Assertion Rule which is LPDDR3 protocol.
3. WL (Write Latency) is the delay, in clock cycles, between the rising edge of SCK where a write command is referenced and the first rising strobe edge where the first byte of write data is present. The WL value is determined by the value of the CL (CAS Latency) setting.
4. The system memory clock outputs are differential (CK and CK#), the CK rising edge is referenced at the crossing point where CK is rising and CK# is falling.
5. The system memory strobe outputs are differential (DRAM_DQS and DRAM_DQS#), the DRAM_DQS rising edge is referenced at the crossing point where DRAM_DQS is rising and DRAM_DQS# is falling, and the DRAM_DQS falling edge is referenced at the crossing point where DRAM_DQS is falling and DRAM_DQS# is rising.
6. When the single ended slew rate of the input Data or Strobe signals, within a byte group, are below 2.5V/nS the tSU and tHD specifications must be increased by a derating factor. The input single ended slew rate is measured from DC to AC levels; VIL_DC to VIH_AC for rising edges, and VIH_DC to VIL_AC for falling edges. No derating is required for single ended slew rates equal to or greater than 2.5V/ns.
7. tSU/tHD measurement made with SR of 2.5V/ns for input single-ended data and strobe.
8. Sampled data according to JEDEC requirement.
9. Maximum Vpp noise on V1p5_ddr_ck was 80mV (20MHz band width limited); maximum Vpp noise on DRAM_VDD_F2 was 90mV (20MHz band width limited).
10. All tDVA/tDVB numbers contributions from Si, package and channel.

Figure 55. LPDDR3 DRAM_DQ Setup/Hold Relationship to/from DRAM_DQSP/DQSN (Read Operation)

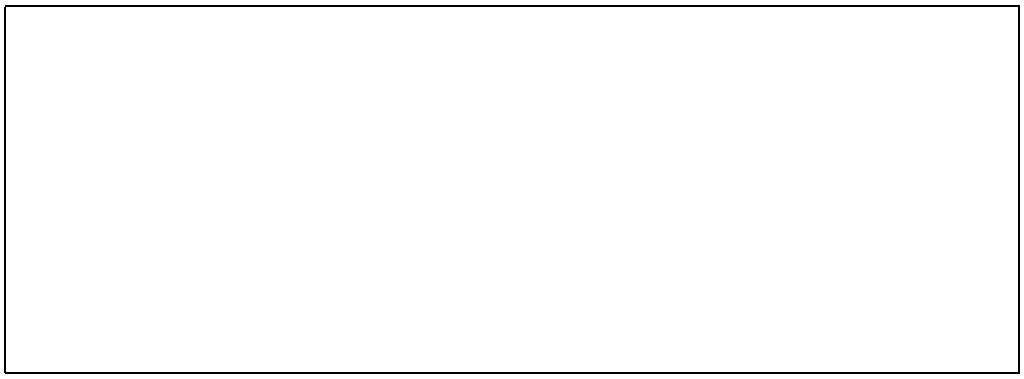


Figure 56. LPDDR3 DRAM_DQ and DRAM_DM Valid before and after DRAM_DQSP/DQSN (Write Operation)

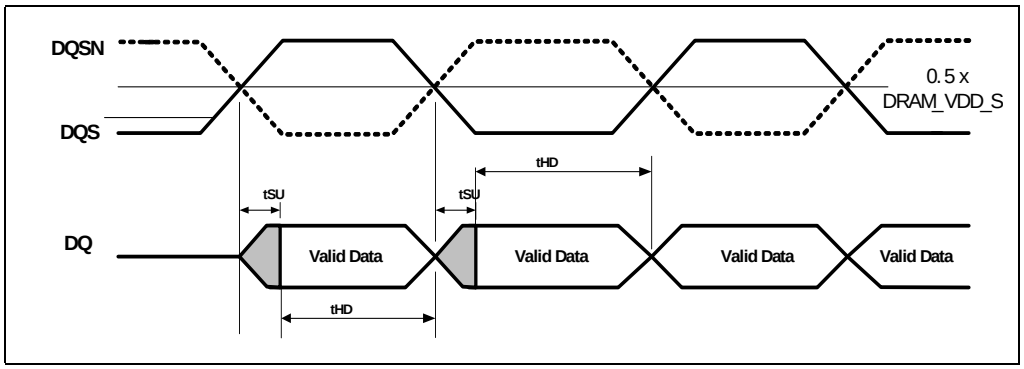


Figure 57. LPDDR3 Write Pre-amble Duration

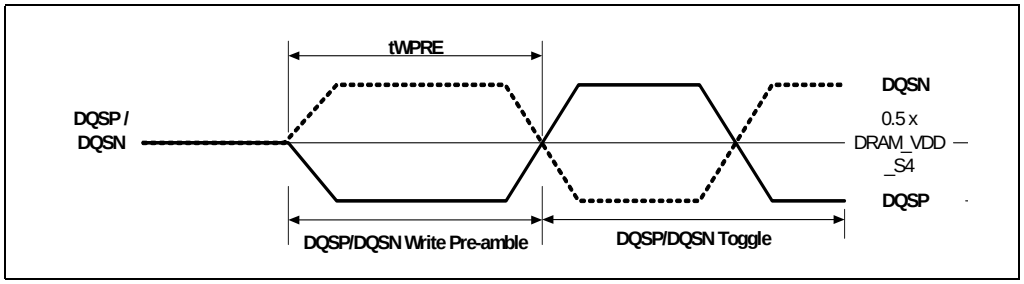


Figure 58. LPDDR3 Write Post-amble Duration

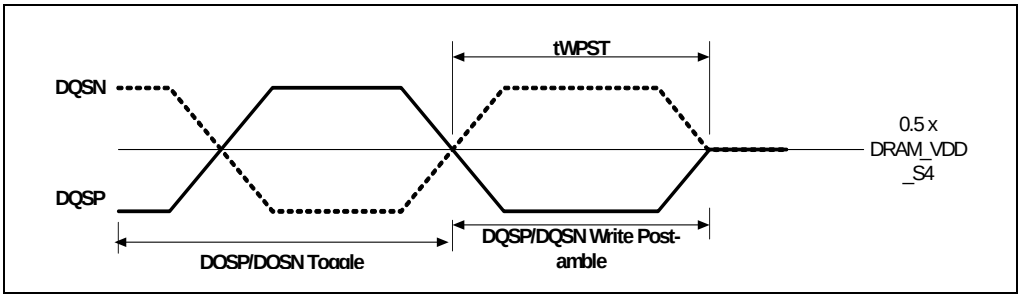


Figure 59. LPDDR3 Command Signals Valid before and after CK Rising Edge

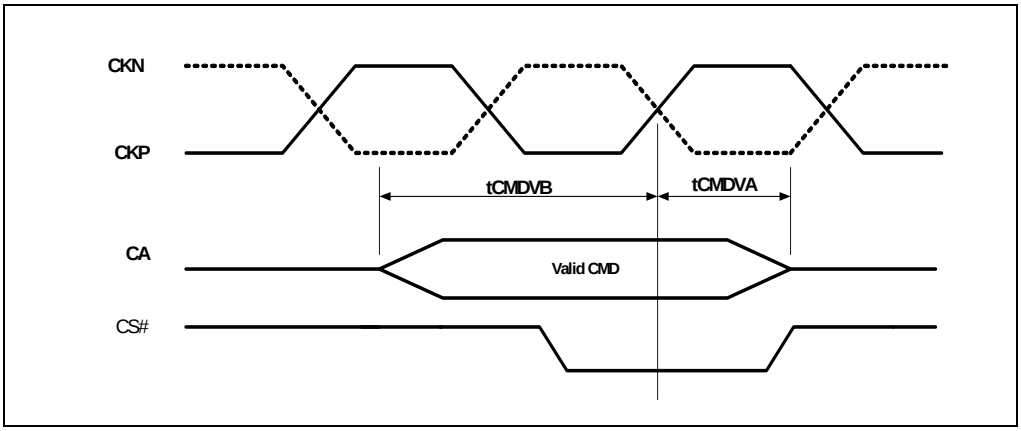


Figure 60. LPDDR3 DRAM_CKE Valid before and after CK Rising Edge

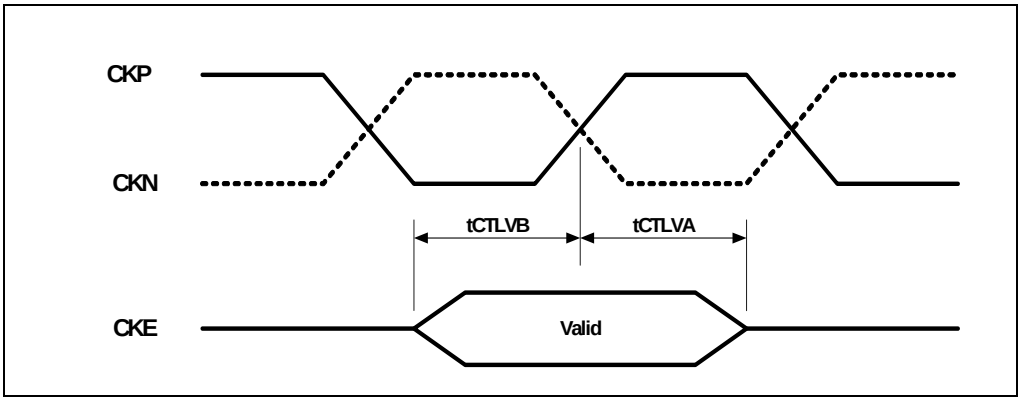


Figure 61. LPDDR3 DRAM_CS# Valid before and after CK Rising Edge

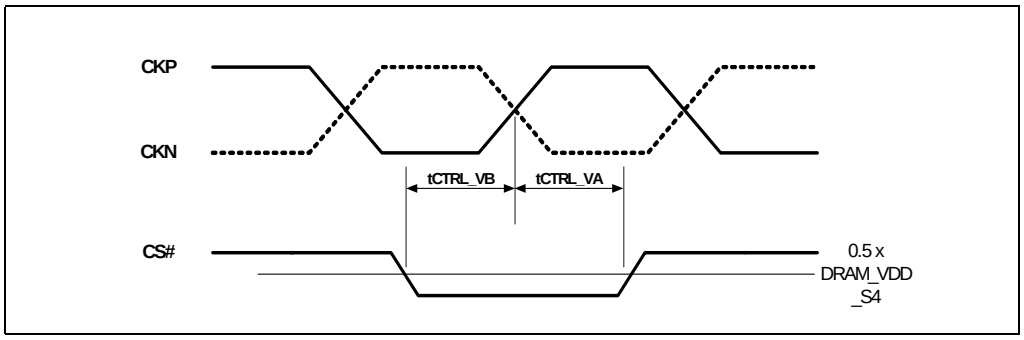


Figure 62. LPDDR3 DRAM_ODT Valid before CK Rising Edge

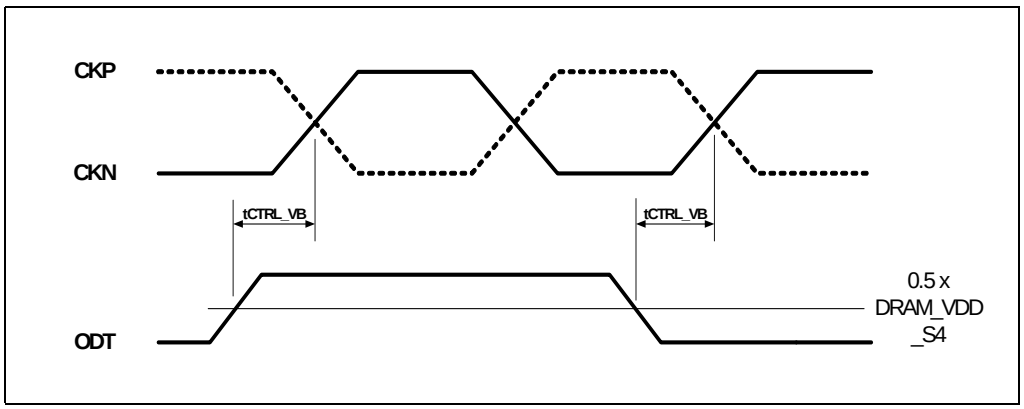


Figure 63. LPDDR3 Clock Cycle Time

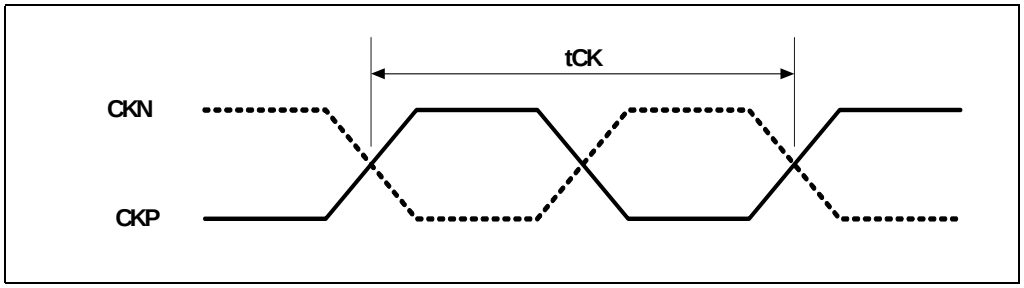
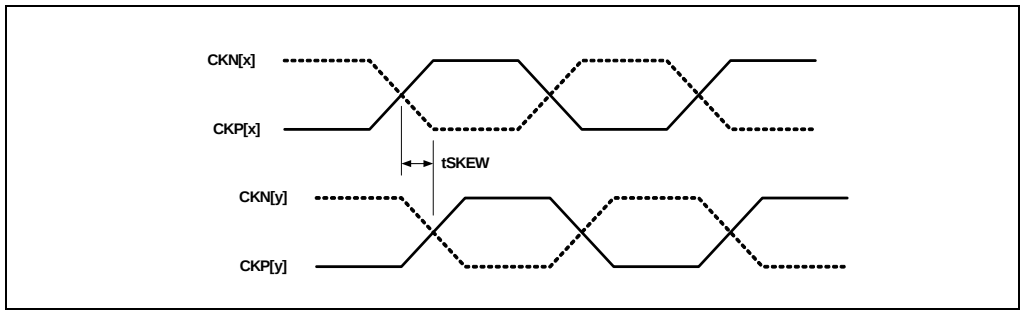


Figure 64. LPDDR3 Skew between System Memory Differential Clock Pairs (DRAM_CKP/CKN)



Note: x represents one differential clock pair, and y represents another differential clock pair within same channel.

Figure 65. LPDDR3 CK High Time

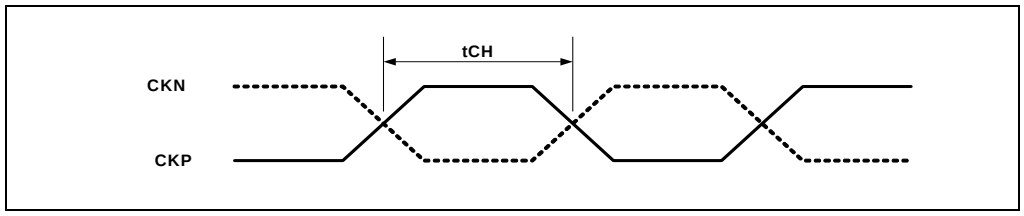


Figure 66. LPDDR3 CK Low Time

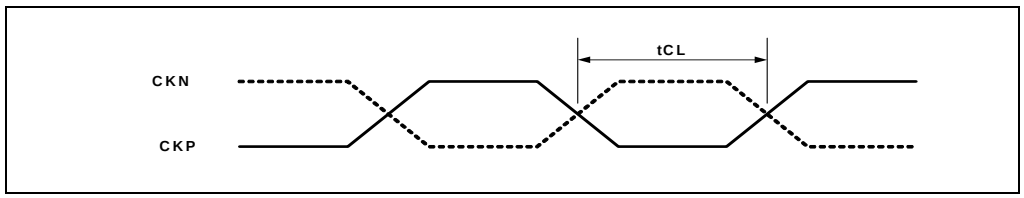


Figure 67. LPDDR3 DRAM_DQS Falling Edge Output Access Time to CK Rising Edge

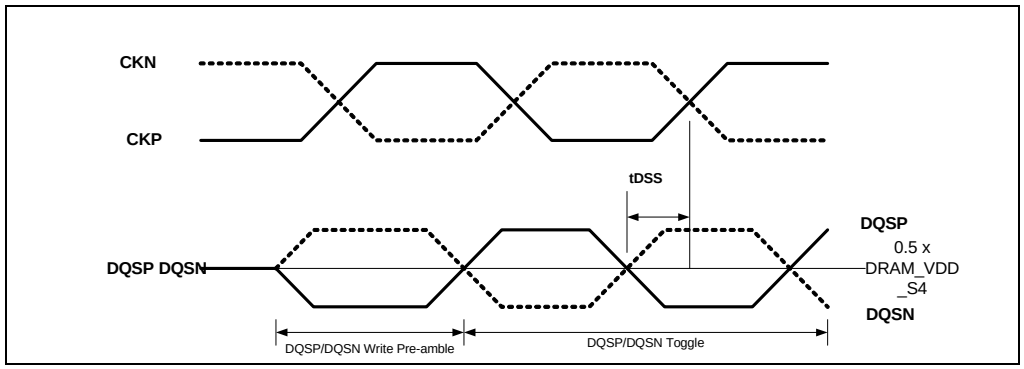


Figure 68. LPDDR3 DRAM_DQS Falling Edge Output Access Time From CK Rising Edge

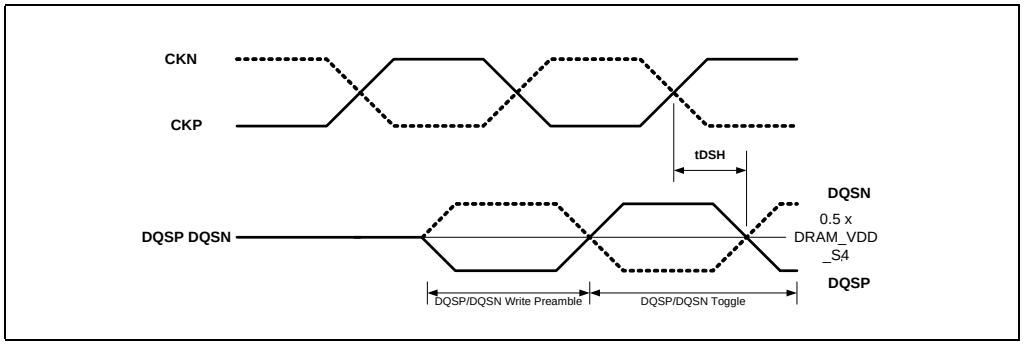
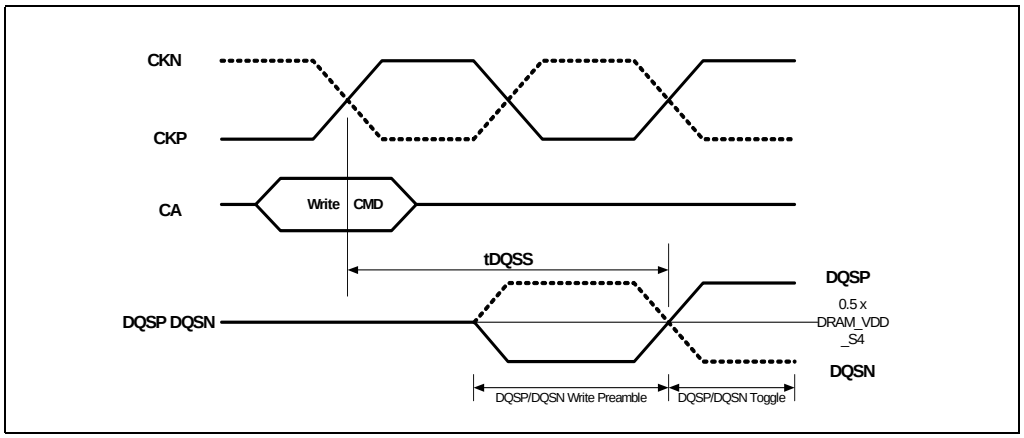


Figure 69. LPDDR3 CK Rising Edge Output Access Time to the 1st DRAM_DQS Rising Edge



20.7.5 Display AC Specifications

20.7.5.1 HDMI AC specification

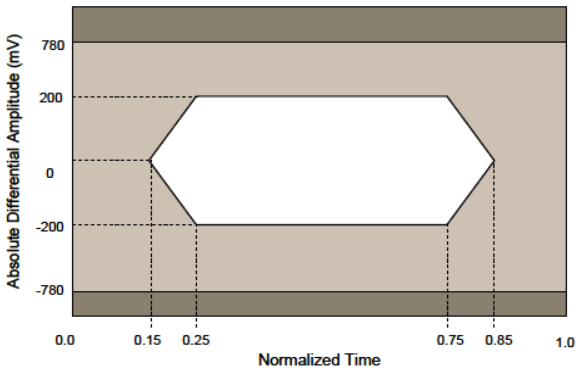
Table 156. HDMI AC specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
T_{RISE}/T_{FALL}	Rise time/fall time (20-80%)	-	75		ps	1 / Fig51
$L_{TX-SKEWINTER\_PAIR}$	Lane-to-Lane Output Skew at Source connector			0.20 $T_{character}$		$T_{character} = 10*T_{bit}$
$L_{TX-SKEWINTRA\_PAIR}$	Lane Intra-pair Output Skew at Source connector			0.15 T_{bit}		
	Clock duty Cycle	40	50	60	%	
	TMDS Differential Clock Jitter			0.25 T_{bit}		

NOTES:

1. 75psec =< Rise time / fall time

Figure 70. Eye Diagram mask for HDMI



20.7.5.2 Display Port Transmitter AC specification

Table 157. Display Port Transmitter AC specification (Sheet 1 of 2)

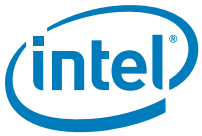
Symbol	Parameter	Min	Typ	Max	Units	Notes
f_{HBR2}	Frequency for High Bit Rate 2	5.37138	5.5	5.40162	Gbps	1
f_{HBR}	Frequency for High Bit Rate	2.68569	2.7	2.70081	Gbps	1
f_{RBR}	Frequency for Reduced Bit Rate	1.61141	1.62	1.620048	Gbps	1
UI_HBR2	Unit Interval for high bit rate 2 (5.4 Gbps / lane)		185		ps	
UI_HBR	Unit Interval for high bit rate (2.7 Gbps / lane)		370		ps	
UI_RBR	Unit Interval for high bit rate (1.62 Gbps / lane)		617		ps	
Down_Spread_Amplitude	Link clock down Spreading	0		0.5	%	2
Down_Spread_Frequency	Link Clock down Spreading Frequency	30		33	kHz	3
C_{TX}	AC Coupling Capacitor	75		200	nF	11
$L_{TX-SKEWINTER\_PAIR\_HBR\_RBR}$	Lane-to-Lane Output Skew			2	UI	
$L_{TX-SKEWINTER\_PAIR\_HBR2}$	Lane-to-Lane Output Skew			4UI + 500ps		
$L_{TX-SKEWINTRA\_PAIR}$	Lane Intra-pair Output Skew			30	ps	
$T_{TX-TJ\ 8b10b\ HBR2}$	Maximum TX Total Jitter			0.62	UI	13

Table 157. Display Port Transmitter AC specification (Sheet 2 of 2)

Symbol	Parameter	Min	Typ	Max	Units	Notes
$T_{TX-DJ\ 8b10b\ HBR2}$	Maximum TX Deterministic Jitter			0.49	UI	13
$T_{TX-TJ\ D10.2\ HBR2}$	Maximum TX Total Jitter			0.40	UI	14
$T_{TX-DJ\ D10.2\ HBR2}$	Maximum TX Deterministic Jitter			0.25	UI	14
$T_{TX-RJ\ D10.2\ HBR2}$	Minimum TX Random Jitter			0.23	UI	14
$T_{TX-DIFFp-p\ HBR2}$	TX Differential Peak-to-Peak EYE Voltage	90			mV	15
$T_{TX-DIFFp-p\_RANGE\ HBR2}$	TX Differential Peak-to-Peak EYE Voltage Measurement Range	0.375		0.625	UI	16
$T_{TX-EYE-CHIP\_HBR2}$	Minimum TX Eye Width at Tx package pins	0.73			UI	17
$T_{TX-EYE-MEDIAN-to-MAX-JITTER\_CHIP\_HBR2}$	Maximum time between the jitter median and maximum deviation from the median at Tx package pins			0.135	UI	17
$T_{TX-EYE-CHIP\_HBR}$	Minimum TX Eye Width at Tx package pins	0.72			UI	4
$T_{TX-EYE-MEDIAN-to-MAX-JITTER\_CHIP\_HBR}$	Maximum time between the jitter median and maximum deviation from the median at Tx package pins			0.147	UI	4
$T_{TX-EYE\_CHIP\_RBR}$	Minimum TX Eye Width at Tx package pins	0.82			UI	5
$T_{TX-EYE-MEDIAN-to-MAX-JITTER\ CHIP\_RBR}$	Maximum time between the jitter median and maximum deviation from the median at Tx package pins			0.09	UI	5
$T_{TX-RISE\_CHIP}$, $T_{TX-FALL\_CHIP}$	D+/D- TX Output Rise/Fall Time at Tx package pins	50		130	ps	6
$I_{TX-SHORT}$	TX Short Circuit Current Limit			50	mA	7
$T_{TX-RISE\_FALL\_MISMATCH\_CHIPDIFF}$	Lane Intra-pair Rise-fall Time Mismatch at Tx package pins.			5	%	8
$F_{TX-REJECTION-BW}$	Clock Jitter Rejection Bandwidth			4	MHz	9
$V_{TX-AC-CM\_HBR\_RBR}$	TX AC Common Mode Voltage			20	mV	2
$V_{TX-AC-CM\_HBR2}$	TX AC Common Mode Voltage			30	mV	2
T_{RISE}/T_{FALL}	Rise time/ Fall time (20%-80%)	75		-	ps	
	Clock duty cycle, min/average/max	40	50	60	%	
	TMDS differential Clock Jitter			0.25	UI	

NOTES:

1. Frequency High limit = +300ppm; Low limit = -5300ppm
2. Range: 0% ~ 0.5% when downspread enabled



3. Range: 30 kHz ~33 kHz when downspread enabled.
4. For High Bit Rate.
5. For Reduced Bit Rate.
6. At 20 to 80
7. Total drive current of the transmitter when it is shorted to its ground.
8. Informative. D+ rise to D- fall mismatch and D+ fall to D- rise mismatch.
9. Informative. Transmitter jitter must be measured at source connector pins using a signal analyzer that has a 2nd order PLL with tracking bandwidth of 20MHz (for D10.2 pattern) and damping factor of 1.428.
10. Measured at 1.62 GHz and 2.7 GHz (if supported), within the frequency tolerance range. Time-domain measurement using a spectrum analyzer.
11. All DisplayPort Main Link lanes as well as AUX CH must be AC coupled. AC coupling capacitors must be placed on the transmitter side. Placement of AC coupling capacitors on the receiver side is optional.
12. 0.20* Tcharacter @165MHz
13. For HBR2. Measured at 1E-9 BER using the HBR2 Compliance EYE pattern.
14. For HBR2. Measured at 1E-9 BER using the D10.2 compliance pattern.
15. For HBR2. Measured at 1E-9 BER using the HBR2 Compliance EYE pattern.
16. For HBR2. Uses 0.5 CDF (Cumulative Distribution Function) of the jitter distribution as the 0UI reference point. TX Differential Peak-to-Peak EYE Voltage requirement can be met anywhere within this UI range.
17. For High Bit Rate 2 using a D10.2 pattern.

20.7.5.3 Embedded Display Port AC Specification

Table 103. lists the recommended electrical parameters for the Main-Link Transmitter of an eDP Source device. These values reflect differences from Display Port Transmitter AC specification. See Table 158 for complete electrical parameters.

Table 158. Embedded Display Port AC specification

Symbol	Parameter	Min	Typ	Max	Units	Notes
UI_Rate_1 (RBR)	Unit Interval for 1.62Gbps/lane	-	617.3	-	ps	1
UI_Rate_2	Unit Interval for 2.16Gbps/lane	-	463.0	-	ps	1
UI_Rate_3	Unit Interval for 2.43Gbps/lane	-	411.5	-	ps	1
UI_Rate_4 (HBR)	Unit Interval for 2.7Gbps/lane	-	370.4	-	ps	1
UI_Rate_5	Unit Interval for 3.24Gbps/lane	-	308.6	-	ps	1
UI_Rate_6	Unit Interval for 4.32Gbps/lane	-	231.5	-	ps	1

NOTES:

1. Nominal Unit Interval (UI) does not account for SSC. For constant (non-SSC) frequency, the frequency range is: High limit = +300ppm / Low Limit = -5300ppm.

20.7.5.4 Display Port AUX Channel AC Specification

Table 159. Display Port AUX Channel AC Specification

Symbol	Parameter	Min	Typ	Max	Units	Notes[1]
UI	AUX Unit Interval	0.4	0.5	0.6	µs	1
$T_{AUX-BUS-PARK}$	AUX CH bus park time	10			ns	2
$T_{CYCLE-to-CYCLE\ Jitter}$	Maximum allowable UI variation within a single transaction at connector pins of a transmitting Device			0.08	UI	3
	Maximum allowable variation for adjacent bit times within a single transaction at connector pins of a transmitting Device			0.04	UI	4
f_{FAUX}	FAUX transaction frequency		1		MHz	7
I_{AUX_SHORT}	AUX Short Circuit Current Limit			90	mA	5
C_{AUX}	AC Coupling Capacitor	75		200	nF	6

NOTES:
1. Results in the bit rate of 1Mbps including the overhead of ManchesterII coding.
2. Period after the AUX CH STOP condition for which the bus is parked
3. Equal to 48 ns maximum. The transmitting Device is a Source Device for a Request transaction and a Sink Device for a Reply Transaction
4. Equal to 24 ns maximum.
5. The transmitting Device is a Source Device for a Request transaction and a Sink Device for a Reply Transaction.Total drive current of the transmitter when it is shorted to its ground.
6. The AUX CH AC-coupling capacitor placed on both the DP upstream and downstream devices.
7. Nominal 675Mbps includes overhead of 8B10B coding Nominal UI is 1389ps Frequency tolerance +/- 300ppm

20.7.5.5 MIPI Display Serial Interface (DSI) AC Specification

Table 160. MIPI DSI AC Characteristics (Sheet 1 of 3)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
MIPI DSI HS-Transmitter Mode						
$\Delta V_{CMTX(HF)}$	Common-level variations above 450 MHz	–	–	15	mV_{RMS}	
$\Delta V_{CMTX(LF)}$	Common-level variation between 50–450 MHz	–	–	25	mV_{PEAK}	
t_R and t_F	20%–80% rise time and fall time	–	–	0.3	UI	
		150	–	–	ps	
$T_{SKEW[TX]}$	Data to Clock Skew [measured at transmitter]	-0.15	–	0.15	UI	1
UI INST	UI Instantaneous (In 1 or 2 or 3 or 4 Lane configuration)	1 (1 Gbps/ 500Mhz)	–	3 (333 Mbps / 167Mhz)	ns	

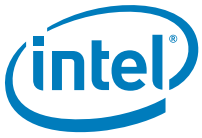


Table 160. MIPI DSI AC Characteristics (Sheet 2 of 3)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
S_{DDTX}	Differential reflection of a Lane Module in High-Speed TX mode				db	Figure 5-63
S_{CCTX}	Common-Mode return loss specification				db	Figure 5-64
MIPI DSI LP-Transmitter Mode						
T_{RLP}/T_{FLP}	15–85% rise time and fall time	–	–	25	ns	
$\delta V/\delta t_{SR}$	Slew rate, C_{LOAD} = 5 pF	–	–	300	mV/ns	5
	Slew rate, C_{LOAD} = 20 pF	–	–	100	mV/ns	5
	Slew rate, C_{LOAD} = 70pF	–	–	150	mV/ns	5
	Slew rate@ C_{LOAD} 5pF to 70pF (Falling Edge Only)	30	–	–	mV/ns	
	Slew rate@ C_{LOAD} 5pF to 70pF (Rising Edge Only)	30	–	–	mV/ns	
	Slew rate@ C_{LOAD} 5pF to 70pF (Rising Edge Only)	30 – 0.075 * ($V_{O,INST}$ – 700)	–	–	mV/ns	
C_{LOAD}	Load Capacitance includes the total interconnect cap load	5	–	70	pF	1
T_{LPX}	Length of any Low-Power state period	–	50	–	ns	2, 3 Figure 72
$T_{LP-PER-TX}$	Period of the LP exclusive-OR clock	–	100	–	ns	4
$T_{LP-PULSE\_TX}$	Pulse width of the LP exclusive-OR clock. First LP exclusive-OR clock puls after Stop State.	40			ns	–
	All other pulses	20			ns	–
Ratio T_{LPX}	Ratio of T_{LPX}(MASTER)/ T_{LPX}(SLAVE) between Master and Slave side	2/3	–	3/2		Figure 72
T_{TA-GET}	Time to drive LP-00 by new TX	$5*T_{LPX}$			ns	Figure 72
T_{TA-GO}	Time to drive LP-00 after Turnaround Request	$4*T_{LPX}$			ns	Figure 72

Table 160. MIPI DSI AC Characteristics (Sheet 3 of 3)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
$T_{TA-SURE}$	Time-out before new TX side starts driving	T_{LPX}		$2*T_{LPX}$	ns	Figure 72
MIPI DSI LP-Receive Mode						
e_{spike}	Input pulse rejection	–	–	300	V*ps	
T_{MIN-RX}	Minimum pulse width response	20	–	–	ns	
V_{INT}	Peak interference amplitude	–	–	200	mV	
f_{INT}	Interference frequency	450	–	–	MHz	

NOTES:
1. Deviates from DPHY specification, which has minimum C_{LOAD} value of 0 pF.
2. T_{LPX} is an internal state machine timing reference. Externally measured values may differ slightly from the specified values due to asymmetrical rise and fall times.
3. MIPI DPHY Revision 1.0 specification states minimum, T_{LPX} = 50 ns. T_{LPX} is not configurable on the SoC.
4. MIPI DPHY Revision 1.0 specification states minimum, $T_{LP-PER-TX}$ = 90 ns. $T_{LP-PER-TX}$ is not configurable on the SoC.
5. DSI LP TX slew rates in EDS spec are not measurable with 0 and 5pf load due capacitance of PCB traces.This issue also mentioned in MIPI Dphy CTS.

Note: The MIPI data-clock TX TSKEW specification also defines the worst case data transition before and after the MIPI clock edge. The maximum allowable TSKEW is 0.15 UI, which dictates that the minimum data transition time before and after MIPI clock edge should be 0.35 UI.

Figure 71. MIPI DSI to Data Clock Timings

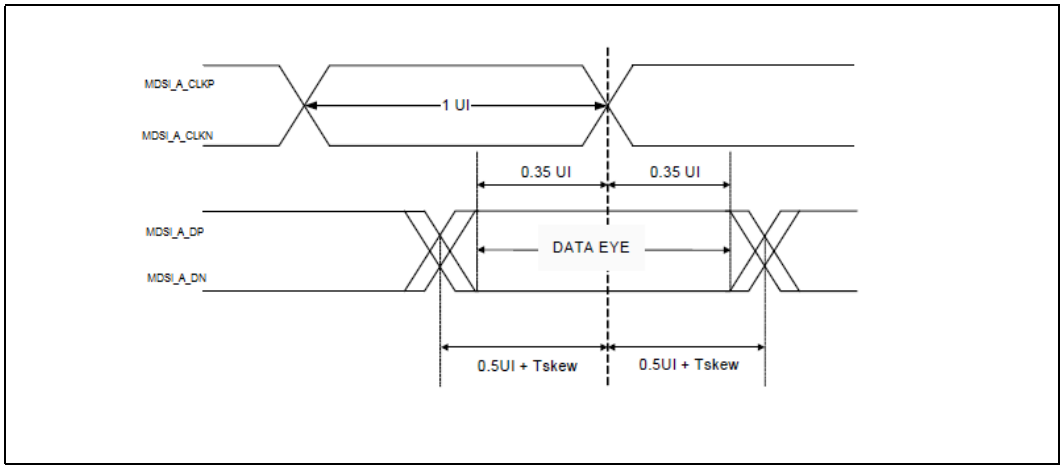
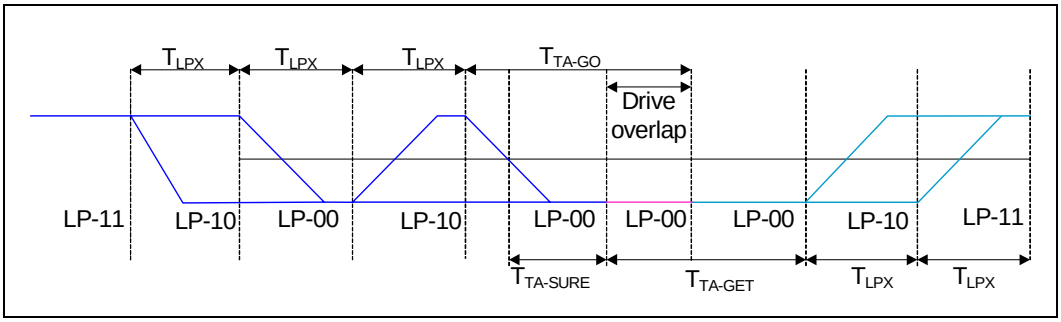


Figure 72. Turnaround Procedure



20.7.6 MIPI-Camera Serial Interface (CSI) AC Specification

Based on version 2 of the MIPI-CSI specification.

Table 161. MIPI-CSI Receiver Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
MIPI HS-Receiver Mode						
$\Delta V_{CMRX(HF)}$	Common-mode interference above 450 MHz	–	–	100	mV	2, 9
$\Delta V_{CMTX(LF)}$	Common-mode interference between 50–450 MHz	-50	–	50	mV	1, 4
C_{CM}	Common-mode termination	–	–	60	pF	3
S_{CDRX}	differential to common-mode	–	–	-26	dB	From 0 to fMAX (2Ghz)
MIPI LP-Receiver Mode						
e_{spike}	Input pulse rejection	–	–	300	V*ps	5, 6, 7
T_{MIN-RX}	Minimum pulse width response	20	–	–	ns	8
V_{INT}	Peak interference amplitude	–	–	200	mV	
f_{INT}	Interference frequency	450	–	–	MHz	

NOTES:
1. Excluding static ground shift of 50 mV.
2. $\Delta V_{CMRX(HF)}$ is the peak amplitude of a sine wave superimposed on the receiver inputs.
3. For higher bit rates a 14 pF capacitor is needed to meet the common-mode return loss specification.
4. Voltage difference compared to the DC average common-mode potential.
5. Time-voltage integration of a spike above V_{IL} when in the LP-0 state or below V_{IH} when in the LP-1 state.
6. An impulse spike less than this will not change the receiver state.
7. In addition to the required glitch rejection, designers shall ensure rejection of known RF-interference.
8. An input pulse greater than this will toggle the output
9. Improves on DPHY specification, which requires 100 mV maximum.

Figure 73. Input Glitch Rejection of Low-Power Receivers

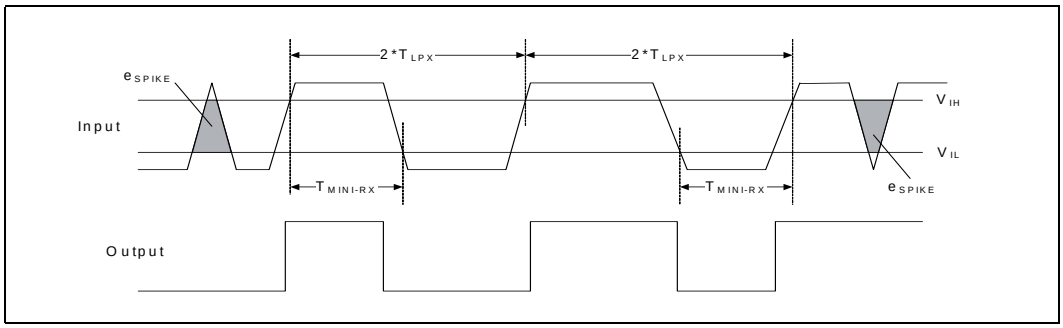


Table 162. MIPI-CSI Clock Signal Specification

Symbol	Clock Parameter	Min.	Typ.	Max.	Unit	Notes
UI_{INST}	UI Instantaneous (In 1 or 2 or 3 or 4 Lane configuration)	0.667 (1.5Gbps/ 750Mhz)		25 (80 Mbps/40Mhz)	ns	1

NOTE: [1]The minimum UI shall not be violated for any single bit period, that is, any DDR half cycle within a data burst.

Figure 74. MIPI-CSI Clock Definition

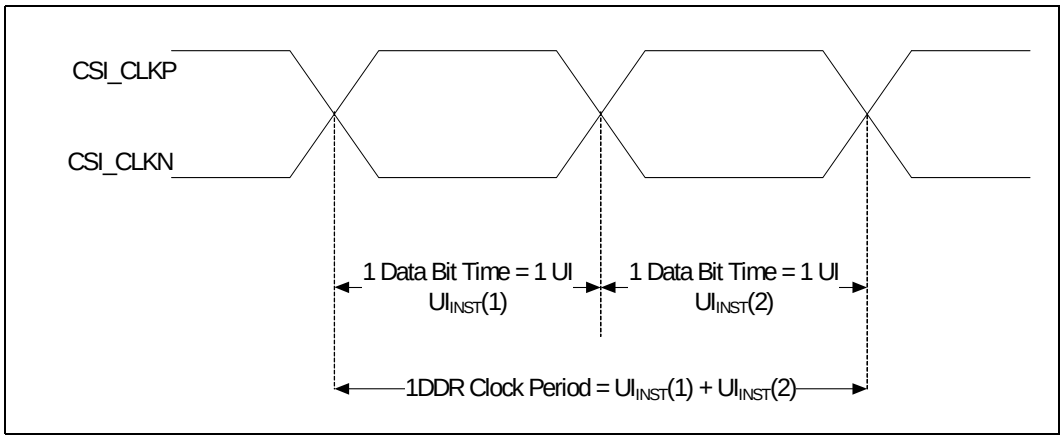


Table 163. MIPI CSI Data Clock Timing Specifications

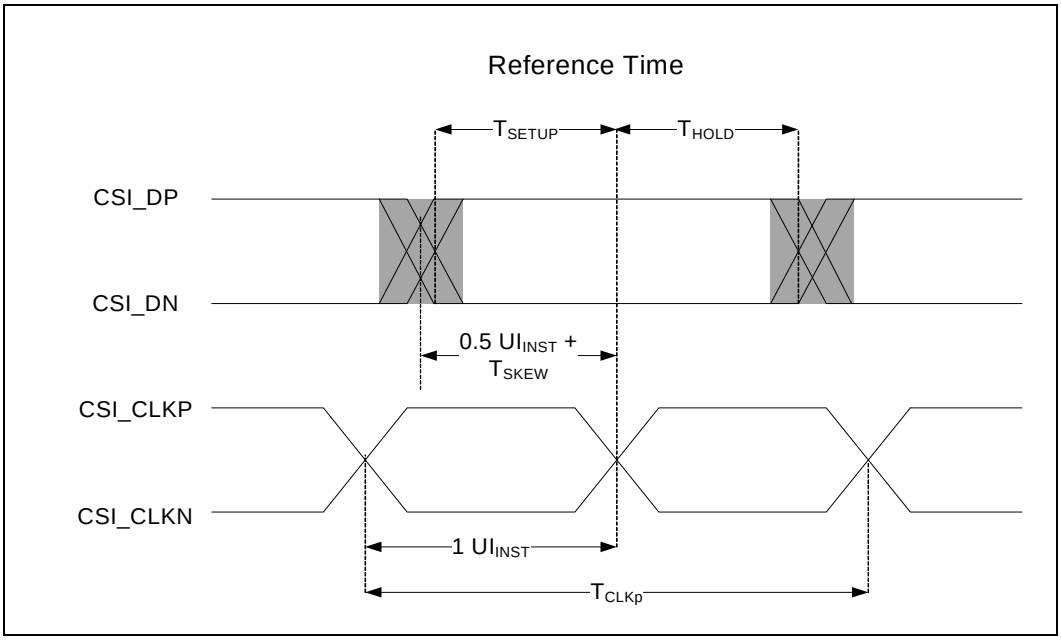
Symbol	Parameter	Min.	Typ.	Max.	Units	Notes
TSETUP[RX]	Data to Clock Setup Time [receiver] Up to 1Gbps	0.15	–	–	UI_{INST}	1, 2

Table 163. MIPI CSI Data Clock Timing Specifications

Symbol	Parameter	Min.	Typ.	Max.	Units	Notes
THOLD[RX]	Clock to Data Hold Time [receiver] Up to 1Gbps	0.15	–	–	UI_{INST}	1, 2
TSETUP[RX]	Data to Clock Setup Time [receiver] after 1Gbps	0.2	–	–	UI_{INST}	1, 2
THOLD[RX]	Clock to Data Hold Time [receiver] after 1Gbps	0.2	–	–	UI_{INST}	1, 2

NOTES:
1. Total silicon and package delay budget of 0.3*UI_{INST}
2. Total setup and hold window for receiver of 0.3*UI_{INST}

Figure 75. MIPI-CSI Data to Clock Timing Definitions



20.7.7 SD Card AC Specification

Table 164. SD Card AC Specification (Sheet 1 of 3)

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
$T_{wc(SDR104)}$	CLK cycle time for SDR104 Mode	-	5	ns	79	2
$T_{wc(SDR50)}$	CLK cycle time for SDR50 Mode	-	10	ns	79	2
$T_{wc(DDR50)}$	CLK cycle time for DDR50 Mode	-	20	ns	79	1

Table 164. SD Card AC Specification (Sheet 2 of 3)

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
$T_{wc(SDR25)}$	CLK cycle time for SDR25 Mode	-	20	ns	80	2
$T_{wc(SDR12)}$	CLK cycle time for SDR12 Mode	-	40	ns	83	2
$T_{ODLY(SDR104)}$	Data Output Delay for SDR104 Mode	0	10	ns	79	2,3
$T_{ODLY(SDR50)}$	Data Output Delay for SDR50 Mode	1.5	7.5	ns	79	2,3
$T_{ODLY(DDR50)}$	Data Output Delay for DDR50 Mode	1.5	7	ns	79	2,3
$T_{ODLY(SDR25)}$	Data Output Delay for SDR25 Mode	1.5	14	ns	80	2,3
$T_{ODLY(SDR12)}$	Data Output Delay for SDR12 Mode	1.5	14	ns	83	2,3
$T_{DVW(SDR104)}$	Data Valid window for SDR104 mode	2.375	-	ns		6
T_{SU_SOC} (SDR)	SoC setup time (data valid before clock launched)	0	–	ns	78 (For SDR50 Mode)	1,4,5
T_{HD_SOC} (SDR)	SoC hold time (data valid after clock launched)	3	–	ns	78 (For SDR50 Mode)	1,4,5
T_{SU_SOC} (DDR)	SoC setup time (data valid before clock launched)	0	–	ns	79 (For DDR50 Mode)	1,4,5
T_{HD_SOC} (DDR)	SoC hold time (data valid after clock launched)	3	–	ns	79 (For DDR50 Mode)	1,4,5
T_{SU_SOC} (SDR)	SoC setup time (data valid before clock launched)	0	–	ns	78 (For SDR12/25 Mode)	1,4,5
T_{HD_SOC} (SDR)	SoC hold time (data valid after clock launched)	3	–	ns	78 (For SDR12/25 Mode)	1,4,5
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (3.3V)	Clock Rise and Fall Time (3.3V operation)	1	4	ns		7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		2	ns	For SDR50 Mode	7

Table 164. SD Card AC Specification (Sheet 3 of 3)

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		4	ns	For DDR50 Mode	7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		10	ns	For SDR12/25 Mode	7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		0.96	ns	For SDR104 Mode	7

1. Timing is measured from 50% to 50% Vdd
2. Timing is measured from 0.975V to 0.975V
3. SoC output timings are measured at SoC pad with a test load of 12pF
4. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
5. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
6. SoC input valid window is measured from minimum of 1.27V to 1.27V and 0.58V to 0.58V
7. SoC output signal rise/fall time spec is measured with test load of 30pF across voltage threshold 35%/ 65%

Figure 76. SD Card Timing Diagram (DDR50)

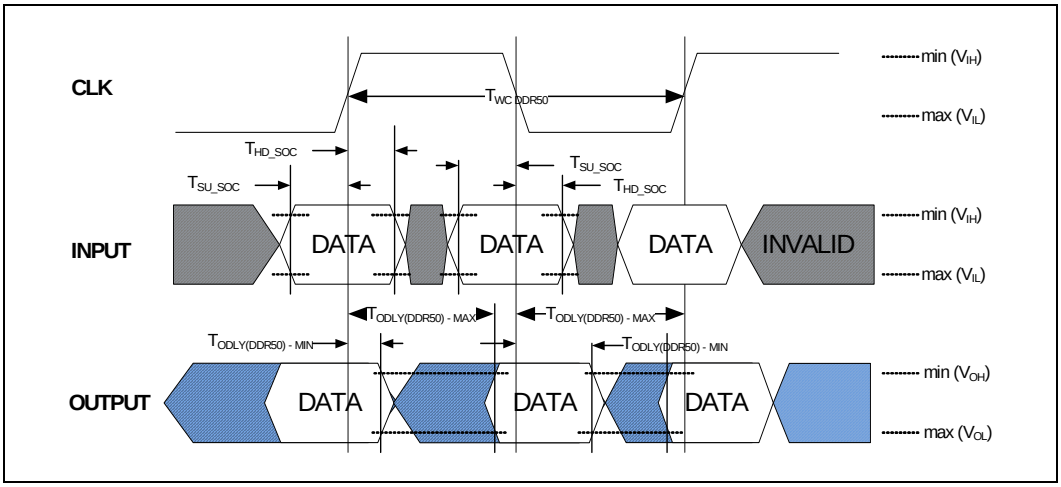


Figure 77. SD Card Output Timing Diagram (SDR25)

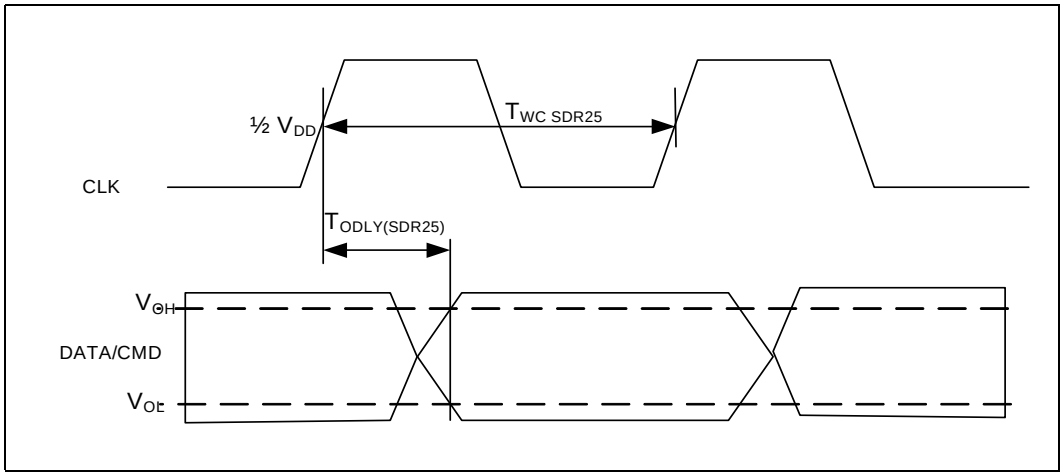
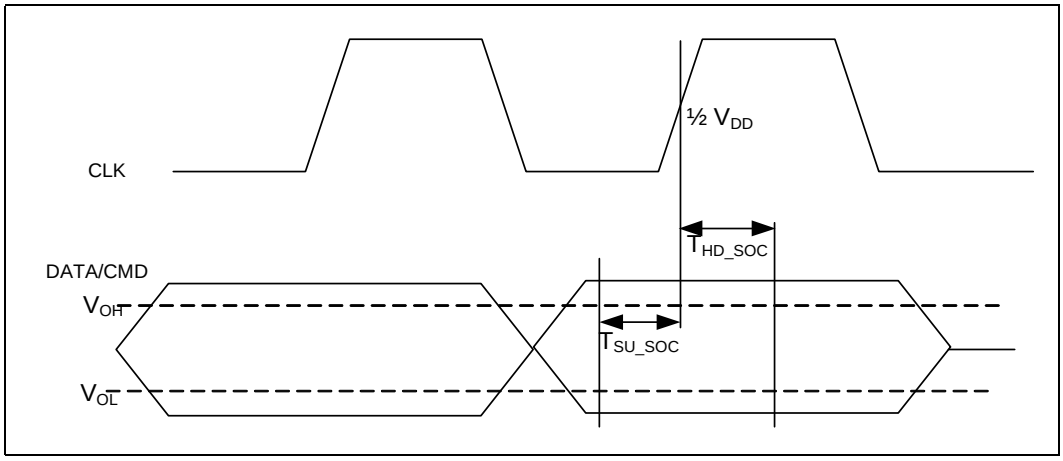


Figure 78. SD Card Input Timing Diagram (SDR12)



20.7.7.1 SD Card Default Speed Specification

Table 165. SD Card Default Speed AC Specification (Sheet 1 of 2)

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
f_{PP}	Clock Frequency Data transfer mode	0	25	MHz		
f_{OD}	Clock Frequency identification mode	0	400	kHz		1
t_{FL}	Clock low time	10	-	ns	Figure 54, 55	
t_{WH}	Clock High time	10	-	ns	Figure 54, 55	
t_{TLH}	Clock Rise time	-	10	ns	Figure 54, 55	
t_{THL}	Clock Fall time	-	10	ns	Figure 54, 55	
	Clock Overshoot	-	4.5	V		

Table 165. SD Card Default Speed AC Specification (Sheet 2 of 2)

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
	Clock undershoot	-1.5	-	V		
Outputs CMD,DAT (referenced to CLK)						
t_{ODLY}	Output Delay time during Data Transfer Mode	0	14	ns	Figure 55	
t_{ODLY}	Output Delay time during Identification Mode	0	50	ns	Figure 55	
Inputs CMD, DAT (referenced to CLK)						
t_{ISU}	Input Set-up time	5	-	ns	Figure 54	
t_{IH}	Input hold time	5	-	ns	Figure 54	

NOTE: 0 Hz means to stop the clock. The given minimum frequency range is for cases were continues clock is required.

Figure 79. SD Card Input Timing Diagram (Default)

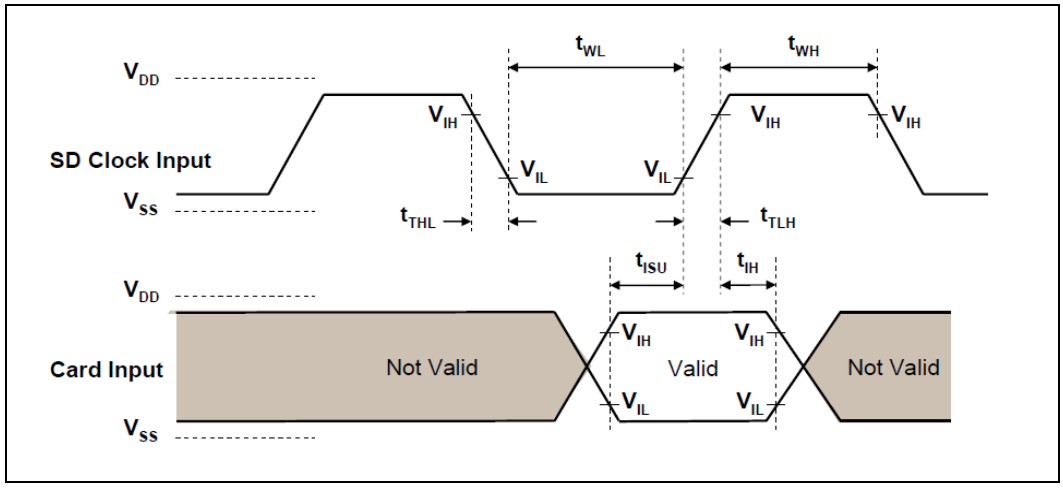
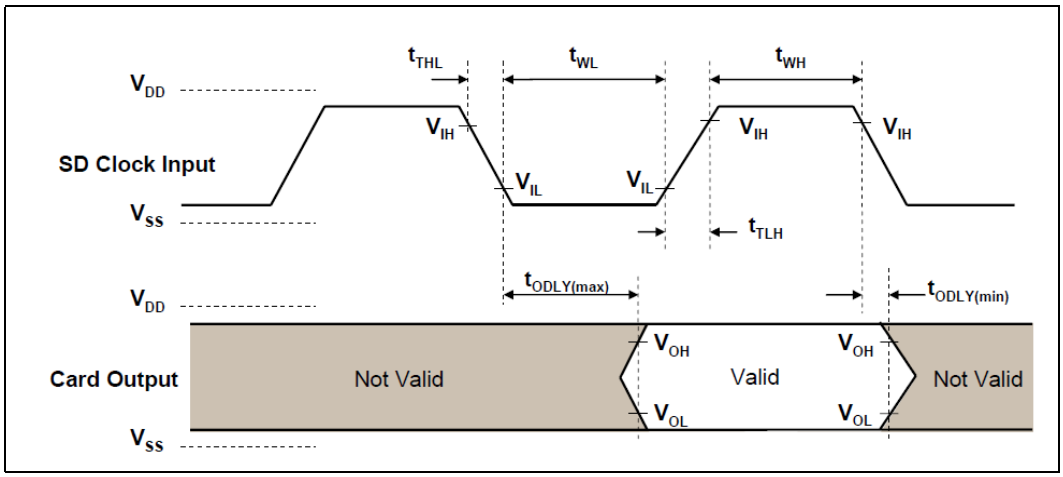


Figure 80. SD card Output Timing Diagram (Default)



20.7.7.2 SD Card High Speed Specification

Table 166. SD Card High Speed AC Specification

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
f_{PP}	Clock Frequency Data transfer mode	0	50	MHz		
t_{FL}	Clock low time	7	-	ns	Figure 56, 57	
t_{WH}	Clock High time	7	-	ns	Figure 56, 57	
t_{TLH}	Clock Rise time	3	-	ns	Figure 56, 57	
t_{THL}	Clock Fall time	3	-	ns	Figure 56, 57	
	Clock Overshoot	-	4.5	V		
	Clock undershoot	-1.5	-	V		
Outputs CMD,DAT (referenced to CLK)						
t_{ODLY}	Output Delay time during Data Transfer Mode	2.5	14	ns	Figure 57	
Inputs CMD, DAT (referenced to CLK)						
t_{ISU}	Input Set-up time	6	-	ns	Figure 56	
t_{IH}	Input hold time	2	-	ns	Figure 56	
C_L	Total Capacitance for each line	-	40	pF		

NOTE: 0 Hz means to stop the clock. The given minimum frequency range is for cases were continues clock is required.

Figure 81. SD Card Input Timing Diagram (High Speed)

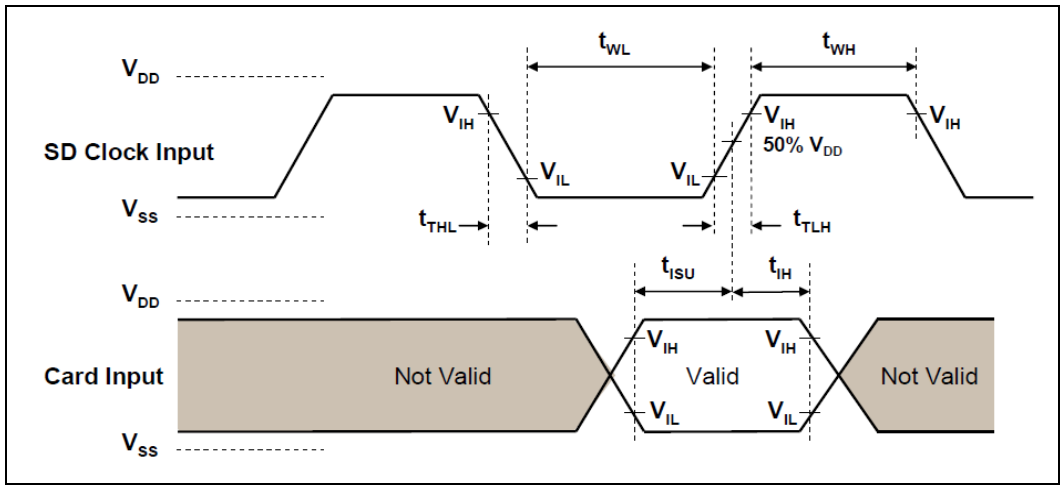
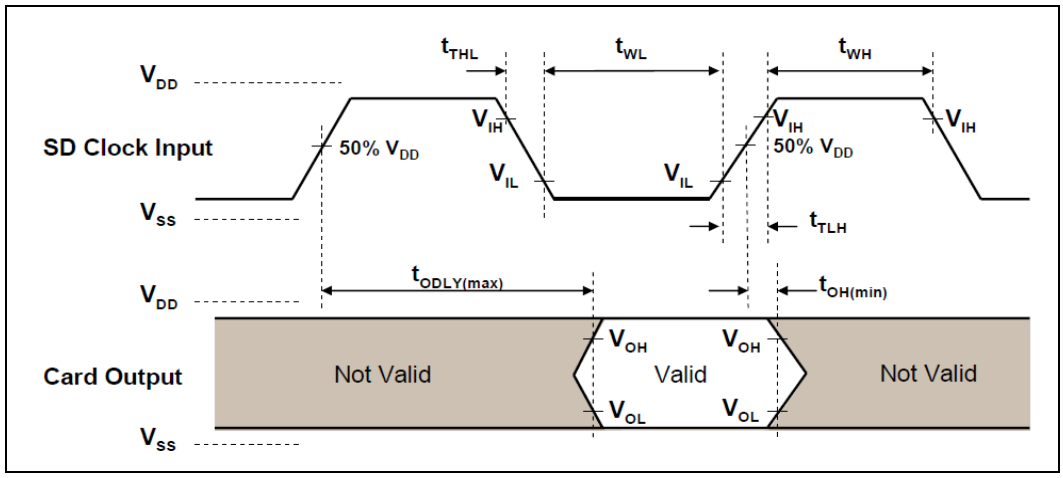


Figure 82. SD card Output Timing Diagram (High Speed)



20.7.8 SDIO AC Specification

Table 167. SDIO AC Specification

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
$T_{wc(SDR104)}$	CLK cycle time for SDR104 Mode	-	5	ns	79	2
$T_{wc(SDR50)}$	CLK cycle time for SDR50 Mode	-	10	ns	79	2
$T_{wc(DDR50)}$	CLK cycle time for DDR50 Mode	-	20	ns	79	1
$T_{wc(SDR25)}$	CLK cycle time for SDR25 Mode	-	20	ns	80	2

Table 167. SDIO AC Specification

Symbol	Parameter	Min.	Max.	Unit	Figure	Notes
$T_{wc(SDR12)}$	CLK cycle time for SDR12 Mode	-	40	ns	83	2
$T_{ODLY(SDR104)}$	Data Output Delay for SDR104 Mode	0	10	ns	79	2,3
$T_{ODLY(SDR50)}$	Data Output Delay for SDR50 Mode	1.5	7.5	ns	79	2,3
$T_{ODLY(DDR50)}$	Data Output Delay for DDR50 Mode	1.5	7	ns	79	2,3
$T_{ODLY(SDR25)}$	Data Output Delay for SDR25 Mode	1.5	14	ns	80	2,3
$T_{ODLY(SDR12)}$	Data Output Delay for SDR12 Mode	1.5	14	ns	83	2,3
$T_{DVW(SDR104)}$	Data Valid window for SDR104 mode	2.375	-	ns		6
T_{SU_SOC} (SDR)	SoC setup time (data valid before clock launched)	0	–	ns	78 (For SDR50 Mode)	1,4,5
T_{HD_SOC} (SDR)	SoC hold time (data valid after clock launched)	3	–	ns	78 (For SDR50 Mode)	1,4,5
T_{SU_SOC} (DDR)	SoC setup time (data valid before clock launched)	0	–	ns	79 (For DDR50 Mode)	1,4,5
T_{HD_SOC} (DDR)	SoC hold time (data valid after clock launched)	3	–	ns	79 (For DDR50 Mode)	1,4,5
T_{SU_SOC} (SDR)	SoC setup time (data valid before clock launched)	0	–	ns	78 (For SDR12/25 Mode)	1,4,5
T_{HD_SOC} (SDR)	SoC hold time (data valid after clock launched)	3	–	ns	78 (For SDR12/25 Mode)	1,4,5
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		2	ns	For SDR50 Mode	7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		4	ns	For DDR50 Mode	7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		10	ns	For SDR12/25 Mode	7
$T_{RISE\ CLK}$/ $T_{FALL\ CLK}$ (1.8V)	Clock Rise and Fall Time (1.8V operation)		0.96	ns	For SDR104 Mode	7

NOTES:
1. Timing is measured from 50% to 50% Vdd

2. Timing is measured from 0.975V to 0.975V
3. SoC output timings are measured at SoC pad with a test load of 12pF
4. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
5. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
6. SoC input valid window is measured from minimum of 1.27V to 1.27V and 0.58V to 0.58V
7. SoC output signal rise/fall time spec is measured with test load of 30pF across voltage threshold 35%/65%

Figure 83. SDIO Timing Diagram (DDR50)

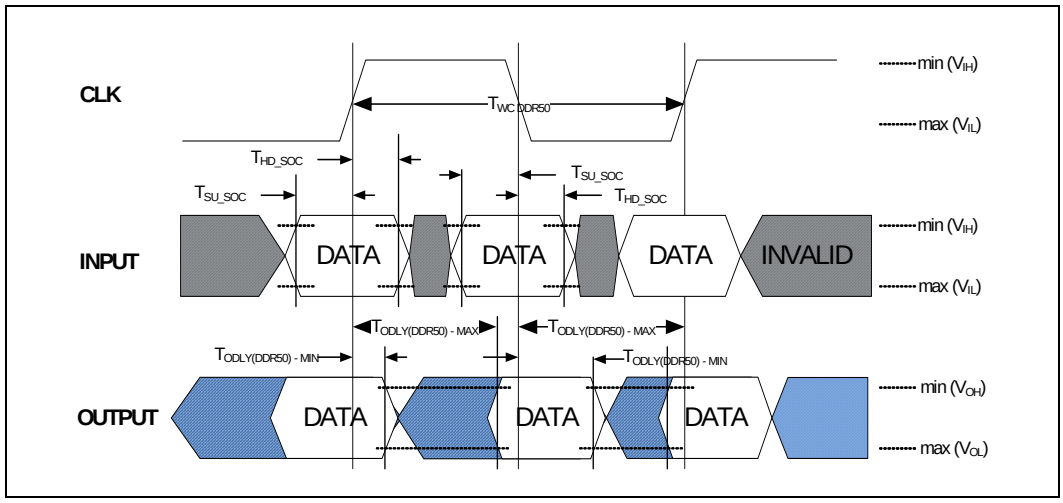


Figure 84. SDIO Output Timing Diagram (SDR25)

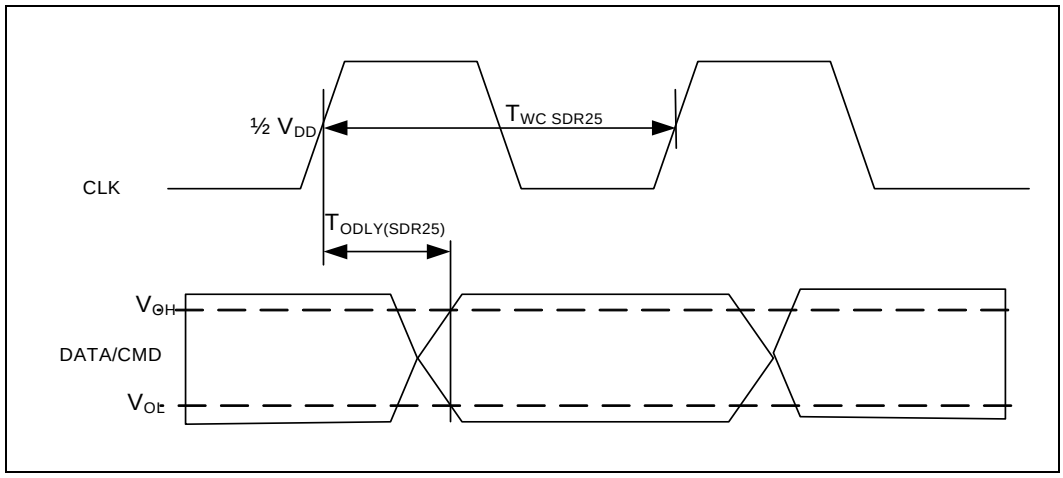


Figure 85. SDIO Output Timing Diagram (SDR12)

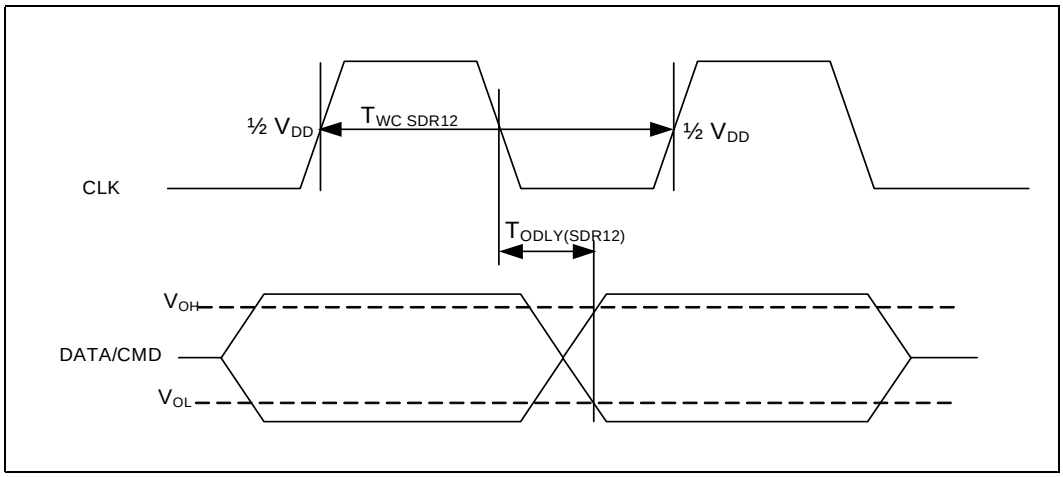
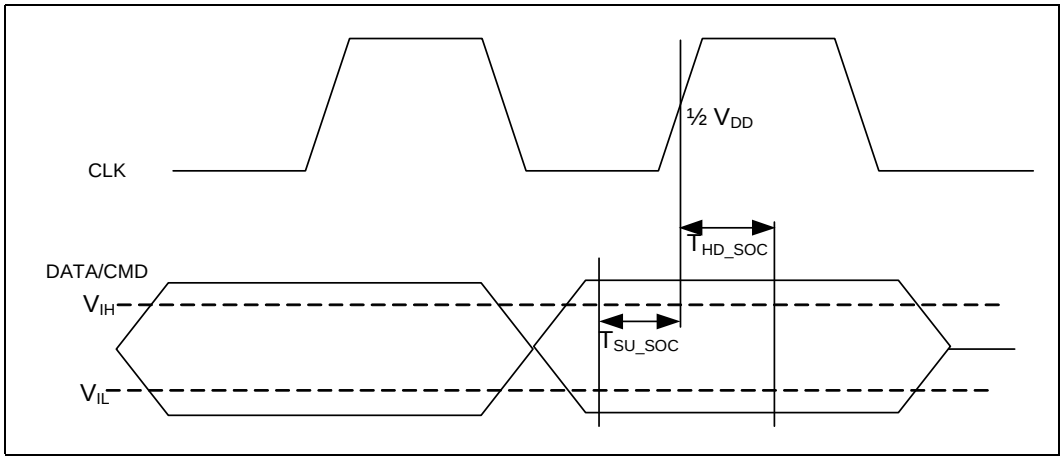


Figure 86. SDIO Input Timing Diagram (SDR12/25)



20.7.9 eMMC 4.51 AC Specification

Table 168. eMMC 4.51 AC Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min.	Max.	Units	Figures	Notes
F_{pp}	Clock Frequency Data transfer Mode	0.4	200	MHZ		1
$T_{wc(BC)}$	CLK Cycle Time (Backward Compatible Mode)	-	40	ns	54	1
$T_{wc(HS/DDR)}$	CLK Cycle Time (High Speed Mode and DDR Modes)	-	20	ns	54	1
$T_{wc(HS200)}$	CLK Cycle Time (HS200 Mode)	-	5	ns	54	1
T_{DC}	CLK Duty Cycle	45	55	%		1

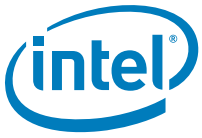


Table 168. eMMC 4.51 AC Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min.	Max.	Units	Figures	Notes
$T_{ODLY(BC)}$	Data/CMD Clock to Output Delay (BC Mode)	8.3	11.7	ns	54	1,3
$T_{ODLY(HS)}$	Data/CMD Clock to Output Delay (HS Mode)	2.5	13.7	ns	55	1,3
$T_{ODLY(DDR50)}$	Data Clock to Output Delay (DDR50 Mode)	1.5	7	ns	55	1,3
$T_{ODLY(HS200)}$	Data Clock to Output Delay (HS200 Mode)	0	10	ns	55	1,3
$T_{SU(BC)}$	eMMC_D Input Setup Time to eMMC_CLK Rising Edge (data read - BC mode)	3	–	ns	56	1,4,5
$T_{H(BC)}$	eMMC_D Input Hold Time to eMMC_CLK Rising Edge (data read - BC mode)	3	–	ns	56	1,4,5
$T_{SU(HS)}$	eMMC_D Input Setup Time to eMMC_CLK Rising Edge (data read - HS mode)	3	–	ns	56	1,4,5
$T_{H(HS)}$	eMMC_D Input Hold Time to eMMC_CLK Rising Edge (data read - HS mode)	3	–	ns	56	1,4,5
$T_{SU(DDR50)}$	eMMC_D Input Setup Time to eMMC_CLK Rising Edge (data read - DDR50 Mode)	2.5	–	ns	55	1,4,5
$T_{H(DDR50)}$	eMMC_D Input Setup Time to eMMC_CLK Rising Edge (data read–DDR50 Mode)	2.5	–	ns	55	1,4,5
T_{DVW}	Data Valid Window	2.375	–	ns		6
$T_{RISE(BC)}$	Rise Time (Output - BC mode)	-	10	ns		7
$T_{FALL(BC)}$	Fall Time (Output -BC mode)	-	10	ns		7
$T_{RISE(HS)}$	Rise Time (Output - HS mode)	-	3	ns		7
$T_{FALL(HS)}$	Fall Time (Output -HS mode)	-	3	ns		7
$T_{RISE(DDR50)}$	Rise Time (Output - DDR50 mode)	-	2	ns		7
$T_{FALL(DDR50)}$	Fall Time (Output - DDR50 mode)	-	2	ns		7
$T_{RISE(HS200)}$	Rise Time (Output - HS200 mode)	-	1	ns		7
$T_{FALL(HS200)}$	Fall Time (Output - HS200 mode)	-	1	ns		7
T_{RSTW}	eMMC_RST# Pulse Width	1	–	µs	57	
T_{RSTCA}	eMMC_RST# to Command Time	200	–	µs	57	
T_{RSTH}	eMMC_RST# High Period (interval time)	1	–	µs	57	

NOTES:
1. Timing is measured from 50% to 50% Vdd
2. Timing is measured from 0.975V to 0.975V.
3. SoC output timings are measured at SoC pad with a test load of 12pF
4. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.

5. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
6. SoC input valid window is measured from minimum of 65% to 65% and 35% to 35%
7. SoC output signal rise/fall time spec is measured with test load of 30pF across voltage threshold 35%/65%

Figure 87. eMMC 4.51 Output Timing Diagram (High Speed Mode)

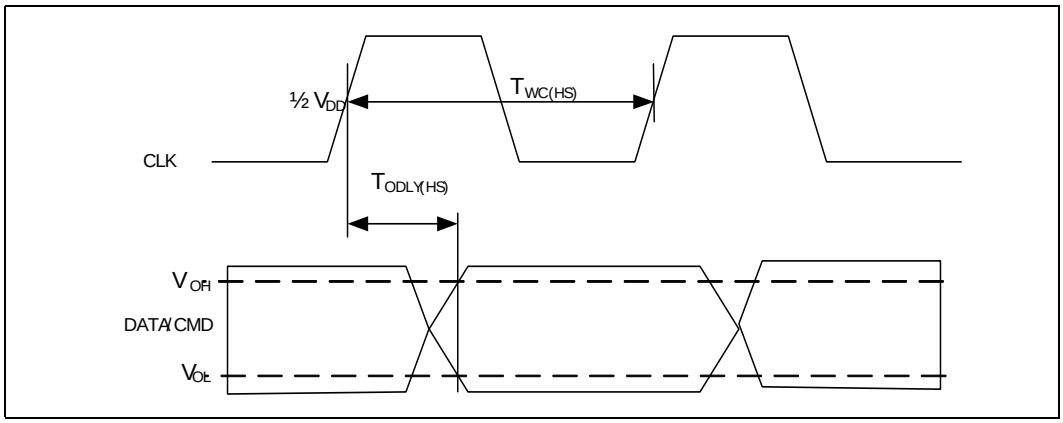


Figure 88. eMMC 4.51 DDR Timings

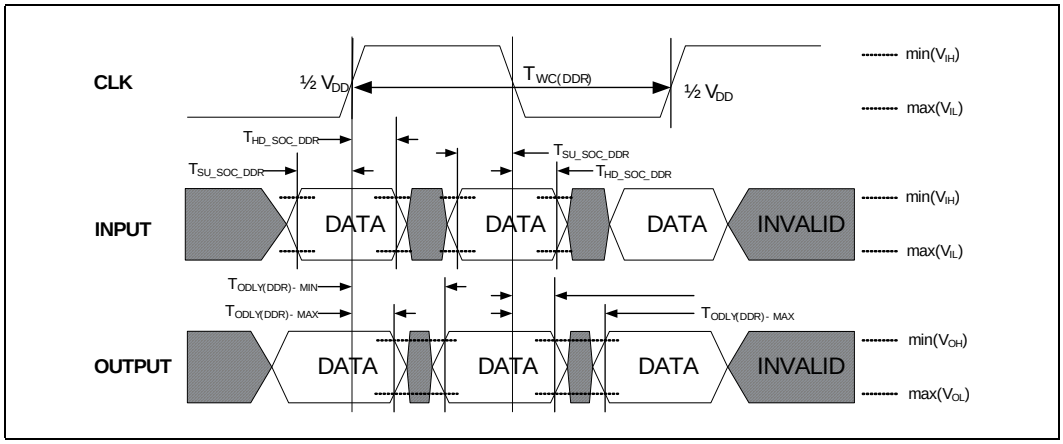


Figure 89. eMMC 4.51 Input Timing Diagram (High Speed Mode)

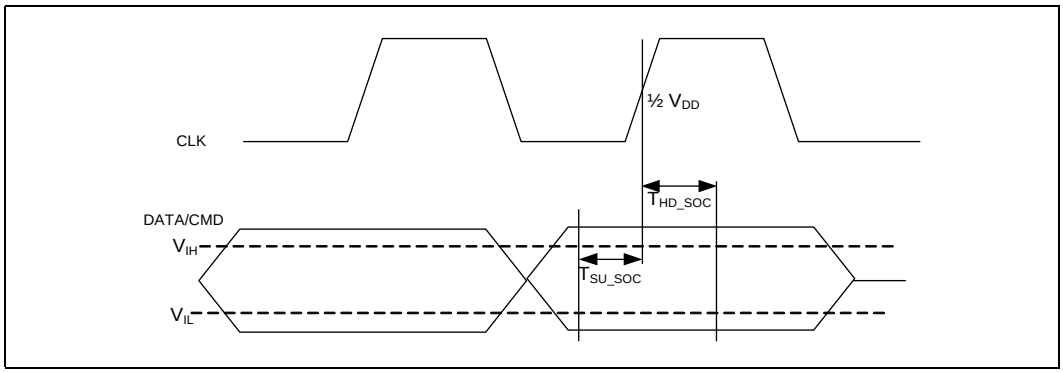


Figure 90. eMMC 4.51 Clock Signal Timing Diagram (HS200 Mode)

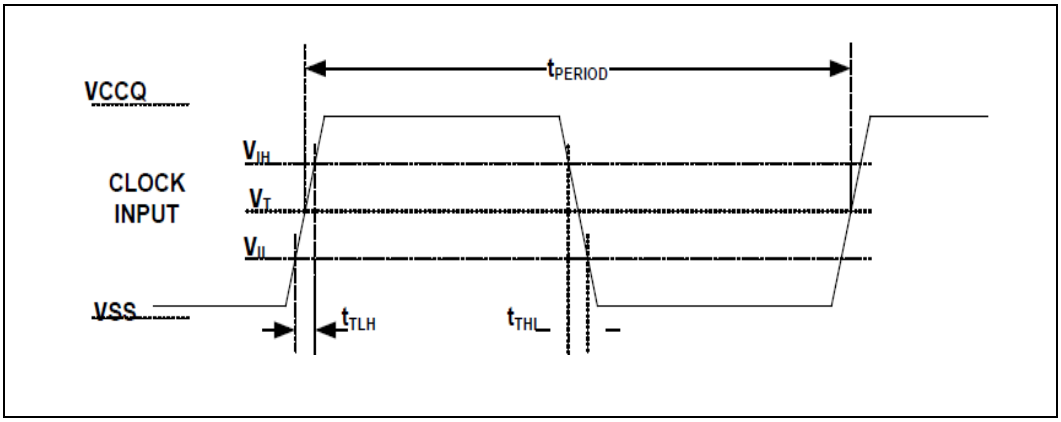
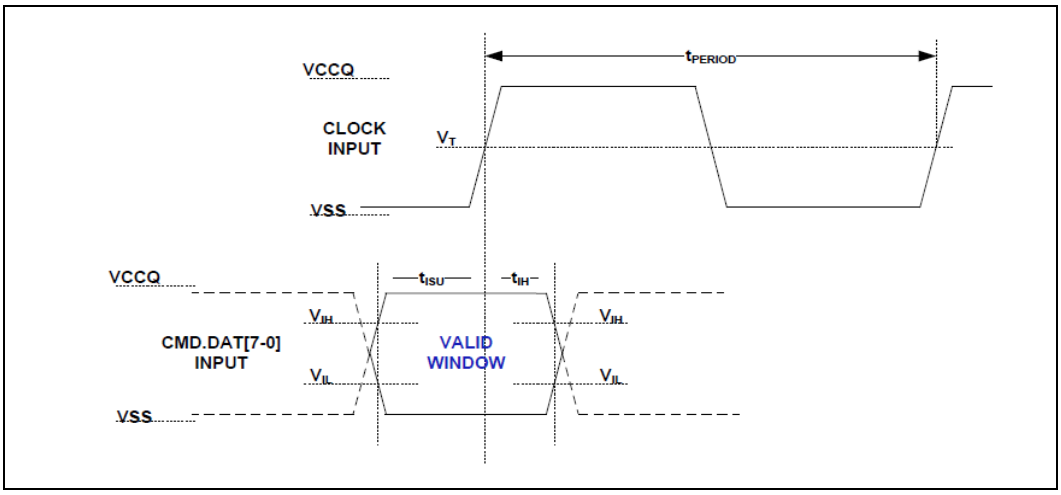


Figure 91. eMMC 4.51 Input Timing Diagram (HS200 Mode)



20.7.10 USB 2.0 Host AC Specification

Table 169. USB 2.0 AC specification (HIGH SPEED)

Symbol	Parameter	Min	Max	Units	Notes	Fig
DRIVER CHARCTERTICS:						
THSR	Rise Time (10% - 90%)	100		ps		
THSF	Fall Time (10% - 90%)	100		ps		
ZHSDRV	Driver Output Resistance (which also serves as high- speed termination)	40.5	49.5			

Table 169. USB 2.0 AC specification (HIGH SPEED)

Symbol	Parameter	Min	Max	Units	Notes	Fig
CLOCK TIMINGS:						
THSDRAT	High-speed Data Rate	479.760	480.240	Mb/s		
THSFRAM	Microframe Interval	124.9375	125.0625	us		

Table 170. USB 2.0 AC specification (FULL SPEED)

Symbol	Parameter	Min	Max	Units	Notes	Fig
DRIVER CHARCTERTICS:						
TFR	Rise Time	4	20	ns		44,45
TFF	Fall Time	4	20	ns		44,45
TFRFM	Differential Rise and Fall Time Matching	90	111.11	%	10	
ZDRV	Driver Output Resistance for driver which is not high-speed capable	28	44			
CLOCK TIMINGS:						
TFDRATHS	Full-speed Data Rate for hubs and devices which are high-speed capable	11.9940	12.0060	Mb/s		
TFDRATE	Full-speed Data Rate for hubs and devices which are not high-speed capable	11.9700	12.0300	MB/s		
TFRAME	Frame Interval	0.9995	1.0005	ms		
FULL-SPEED DATA TIMINGS						
TDJ1 TDJ2	Source Jitter Total (including frequency tolerance): To Next Transition For Paired Transitions	-3.5 -4	3.5 4	ns ns	7,8,12,10	46
TFDEOP	Source Jitter for Differential Transition to SE0 Transition	-2	5	ns	8,11	47
TFEOPT	Source SE0 interval of EOP	160	175	ns		47

Table 171. USB 2.0 AC specification (LOW SPEED)

Symbol	Parameter	Min	Max	Units	Notes	Fig
DRIVER CHARCTERTICS:						
TLR	Rise Time	75	300	ns		44
TFF	Fall Time	75	300	ns		44

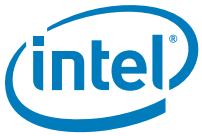


Table 171. USB 2.0 AC specification (LOW SPEED)

Symbol	Parameter	Min	Max	Units	Notes	Fig
TLRFM	Differential Rise and Fall Time Matching	80	125	%	10	
CLOCK TIMINGS:						
TLDRATHS	Low-speed Data Rate for hubs and devices which are high-speed capable	1.49925	1.50075	Mb/s		
TLDRATE	Low-speed Data Rate for hubs and devices which are not high-speed capable	1.4775	1.5225	MB/s		
FULL-SPEED DATA TIMINGS						
TUDJ1 TUDJ2	Upstream facing port source Jitter Total (including frequency tolerance): To Next Transition For Paired Transitions	-95 -150	95 150	ns ns	7,8	46
TDDJ1 TDDJ2	Downstream facing port source Jitter Total (including frequency tolerance): To Next Transition For Paired Transitions	-25 -14	25 14	ns ns	7,8	46
TLDEOP	Source Jitter for Differential Transition to SE0 Transition	-40	100	ns	8,11	47
TLEOPT	Source SE0 interval of EOP	1.25	1.50	us		47
TLST	Width of SE0 interval during differential transition	-	-	210	ns	

NOTES:

1. Measured at A plug.
2. Measured at A receptacle.
3. Measured at B receptacle.
4. Measured at A or B connector.
5. Measured with RL of 1.4 kΩ to 3.6 V.
6. Measured with RL of 14. kΩ to GND.
7. Timing difference between the differential data signals.
8. Measured at crossover point of differential data signals.
9. The maximum load specification is the maximum effective capacitive load allowed that meets the target VBUS drop of 330 mV.
10. Excluding the first transition from the Idle state.
11. The two transitions should be a (nominal) bit time apart.
12. For both transitions of differential signaling.
13. Must accept as valid EOP.
14. Single-ended capacitance of D+ or D- is the capacitance of D+/D- to all other conductors and, if present, shield in the cable. That is, to measure the single-ended capacitance of D+, short D-, VBUS, GND, and the shield line together and measure the capacitance of D+ to the other conductors.
15. For high power devices (non-hubs) when enabled for remote wakeup.

Figure 92. USB Rise and Fall Times

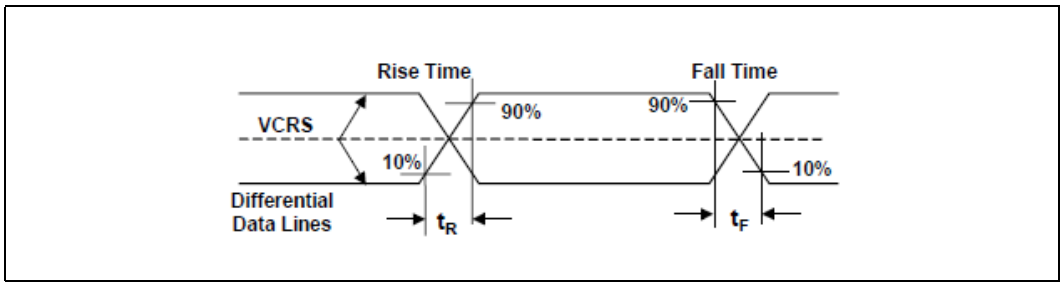


Figure 93. USB Full Speed Load

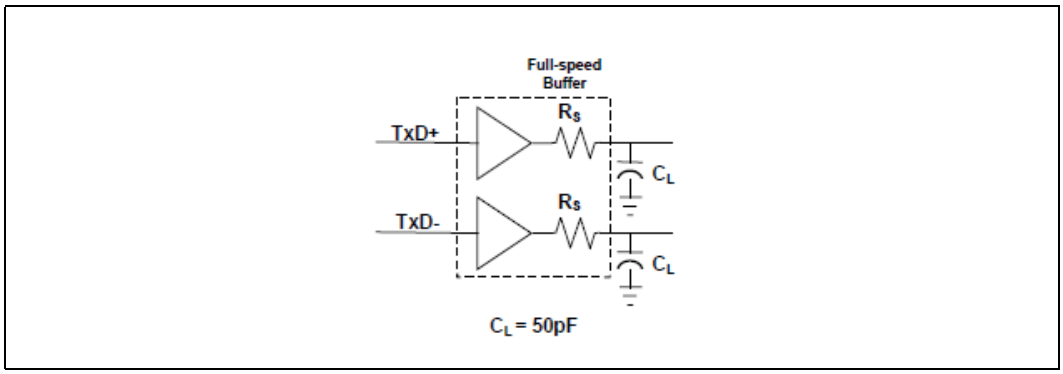


Figure 94. USB Differential Data Jitter for Low / Full- Speed

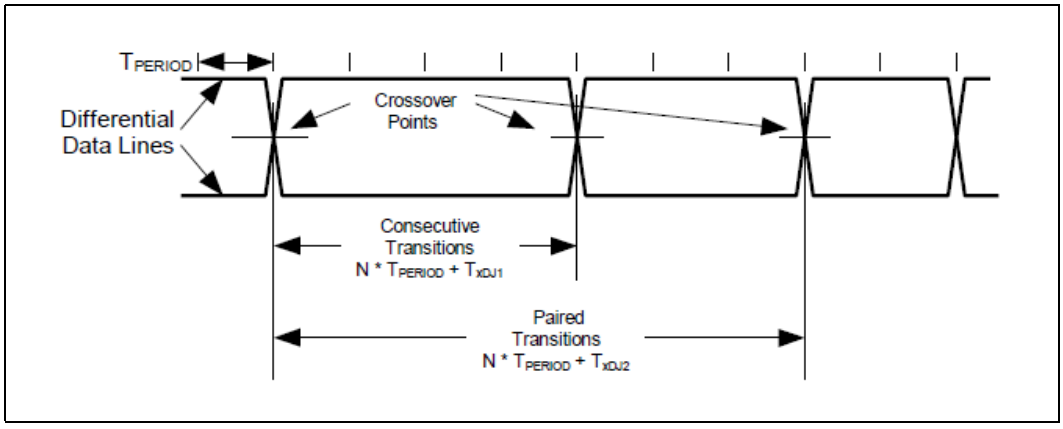
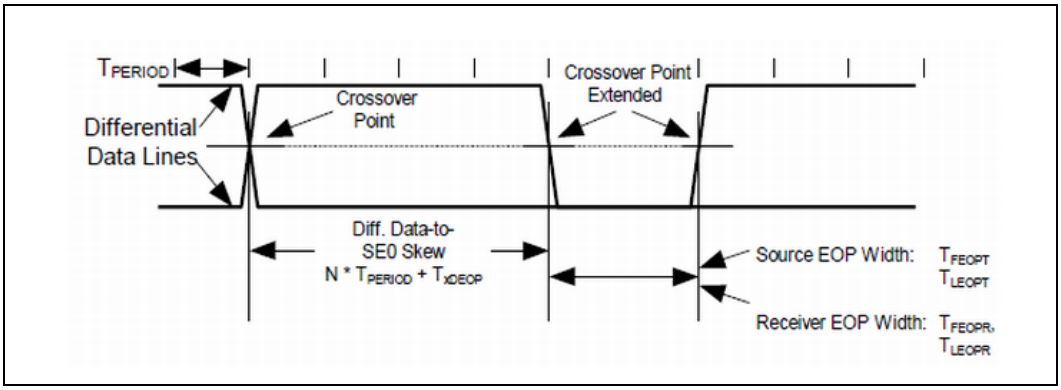


Figure 95. USB Differential-to-EOP Transition Skew and EOP Width for Low/Full-Speed



20.7.11 USB 2.0 HSIC AC Specification

Table 172. USB 2.0 HSIC AC Specification

Symbol	Parameter	Min	Typical	Max	Units	Notes
F_{STROBE}	STROBE Frequency	239.988	240	240.012	MHz	1,2
T_{SLEW}	Slew Rate (rise and fall) Strobe and Data	0.60 * V_{ref}	1.0	1.2	V/ns	1,3
T_S	Setup Time (with respect to STROBE)	300	-	-	ps	1,4
T_H	Hold Time (with respect to STROBE)	300	-	-	ps	1,4

NOTE:
1. +- 500ppm.
2. Jitter and duty cycle are not separately specified parameters, they are incorporated into the values in the table above.
3. Averaged from 30% - 70% points.
4. Measured at the 50% point.

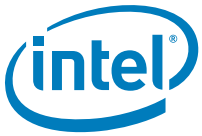
20.7.12 USB 3.0 AC Specification

Table 173. USB 3.0 Signals AC Specification

Symbol	Parameter	Min	Max	Units	Notes
TMIN-PULSE-DJ	Deterministic min Pulse	-	0.96	UI	1
TMIN-PULSE-TJ	Tx min Pulse	-	0.90	UI	2
TTX-EYE	Transmitter Eye	0.625	-	UI	3
TTx-DJ-DD	Tx deterministic jitter	-	.205	UI	4

NOTE:

1. Tx pulse width variation that is deterministic.
2. Min Tx Pulse at 10^{-12} including Dj and Rj.
3. Includes all jitter sources.
4. Deterministic jitter only assuming the Dual Dirac distribution



20.7.13 USB SSIC AC Specification

Table 174. USB SSIC Signals AC Specification

Symbol	Parameter	Min	Max	Units
tPingResponse	Time between device reception of the last framing symbol of a ping and the first framing symbol of the ping_response	-	5000	ns
tNRDYorSTALLResponse	Time between device reception of the last framing symbol for an ACK TP or a DPP or a STATUS TP and the first framing symbol of an NRDY or STALL response	-	5000	ns
tDPResponse	Time between device reception of the last framing symbol for an ACK TP and the first framing symbol of a DP response	-	5000	ns
tACKResponse	Time between device reception of the last framing symbol for a DPP or a STATUS TP and the first framing symbol of an ACK response	-	5000	ns
tMaxBurstInterval	Time between DPs when the device or host is bursting. If the device cannot meet this maximum time, then it shall set the EOB flag in the last DP it sends.	-	1000	ns
tHostACKResponse	Time between host reception of the last framing symbol for a DPP and the first framing symbol of an ACK response	-	5000	ns

20.7.14 I^2S (Audio) AC Specification

Table 175. I^2S Master Mode AC Specifications

Sym	Parameter	Min	Max	Units	Notes	Fig
T_{I2S}	Clock Frequency		9.6	MHz	1	
T_{DC}	Clock Duty Cycle	45	55	%	1	
T_{S_RXD}	Setup for DATAIN with respect to the CLK active edge.	24		ns	1,2,3	
T_{H_RXD}	Hold for DATAIN with respect to the CLK active edge.	0		ns	1,2,3	
T_{CO_TXD}	TCO of DATAOUT with respect to CLK active edge at the SoC	-1	6.5	ns	1,4	
T_{CO_FS}	TCO of FRM with respect to CLK at the SoC	-1.4	7.5	ns	1,4	
$T_{RISE/FALL}$	Minimum and Maximum Rise/Fall Time		1.2	ns	5	

Note: 1. Timing is measured from 50% to 50% Vdd

2. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.

3. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)

4. SoC output timings are measured at SoC pad with a test load of 12pF

5. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 35%/65%

Table 176. I²S Slave Mode AC Specifications

Sym	Parameter	Min	Max	Units	Notes	Fig
T_{I2S}	Clock Frequency		9.6	MHz	1	
T_{DC}	Clock Duty Cycle	47	53			
T_{S_RXD}	Setup for DATAIN with respect to the CLK active edge.	5		ns	1,2,3	
T_{H_RXD}	Hold for DATAIN with respect to the CLK active edge.	3.5		ns	1,2,3	
T_{S_FS}	Setup for FRM with respect to the CLK active edge.	3.5		ns	1,2,3	
T_{H_FS}	Hold for FRM with respect to CLK active edge.	3		ns	1,2,3	
T_{CO_TXD}	Tco of DATAOUT with respect to CLK active edge at the host	7	29	ns	1,4	
$T_{RISE/FALL}$	Minimum and Maximum Rise/Fall Time		1.2	ns	5	

Note: 1. Timing is measured from 50% to 50% Vdd

2. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.

3. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)

4. SoC output timings are measured at SoC pad with a test load of 12pF

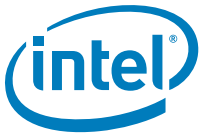
5. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 35%/65%

20.7.15 PMC - Suspended Clock AC Specification

Table 177. SUS Clock Timings

Symbol	Parameter	Min	Max	Units	Notes	Figure
f_{susclk}	Operating Frequency	32		kHz	1	
T_{rise}	Minimum and Maximum Rise Time		2	ns	1	
T_{fall}	Minimum and Maximum Fall Time		2	ns	1	

Note: 1. SoC output signal rise/fall time spec is measured with test load of 30pF across voltage threshold 35%/65%



20.7.16 SPI AC Specification

Table 178. SPI AC Specifications

Sym	Parameter	Min	Max	Units	Notes
T_{freq}	Serial Clock Frequency	-	25	MHz	1,2
T_{DC}	SPI Clock duty cycle at Host	44	55	%	1
$T_{co\text{-}mosi}$	Tco of SPI_MOSI with respect to serial clock edge at the host	0	7.1	ns	1,5
$T_{setup\text{-}miso}$	Setup of SPI_MISO with respect to serial clock edge at the host	13.6		ns	3,4
$T_{hold\text{-}miso}$	Hold of SPI_MISO with respect to serial clock falling edge at the host	0		ns	3,4
$T_{setup\text{-}cs}$	Setup of SPI_CS[1:0]# with respect to serial clock edge at the host	20		ns	1,4
$T_{hold\text{-}cs}$	Hold of SPI_CS[1:0]# with respect to serial clock at the host	10		ns	1,4
$T_{idle\text{-}cs}$	Min Idle (de-assertion) time for SPI_CLK signals	20		ns	1
Trise/Tfall	Rise / Fall time		2	ns	6

NOTES:
1. Timing is measured from 50% to 50% Vdd
2. Device's Tco should be less than 25ns to support 25MHz
3. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
4. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
5. SoC output timings are measured at SoC pad with a test load of 12pF

6.SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 0.58V/1.27V

20.7.17 PCU- Fast SPI AC Specification

Table 179. Fast SPI AC Specifications (Sheet 1 of 2)

Sym	Parameter	Min	Max	Units	Notes
T_{freq}	Serial Clock Frequency	14.28	25	MHz	1
T_{DC}	SPI Clock duty cycle at Host	45	55	%	1
T_{co}	Tco of FST_SPI_D[3:0] with respect to serial clock edge at the host	0	5.2	ns	1,4
T_{setup}	Setup of FST_SPI_D[3:0] with respect to serial clock edge at the host	8.8		ns	1,2,3
T_{hold}	Hold of FST_SPI_D[3:0] with respect to serial clock falling edge at the host	0		ns	1,2,3
$T_{setup\text{-}cs}$	Setup of FST_SPI_CS[2:0]_N with respect to serial clock edge at the host	40		ns	1

Table 179. Fast SPI AC Specifications (Sheet 2 of 2)

Sym	Parameter	Min	Max	Units	Notes
$T_{hold-cs}$	Hold of FST_SPI_CS[2:0]_N with respect to serial clock at the host	40		ns	1
T_{idle}	Min Idle (de-assertion) time for FST_SPI_CLK signals	40		ns	1
Trise/ Tfall	Rise / Fall time		2	ns	5

NOTES:
1. Timing is measured from 50% to 50% Vdd
2. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
3. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
4. SoC output timings are measured at SoC pad with a test load of 12pF
5. SoC output signal rise/fall time spec is measured with test load of 30pF across voltage threshold 35%/ 65%.

20.7.18 PCU - LPC AC Specification

Table 180. LPC AC Specifications

Sym	Parameter	Min	Max	Units	Notes	Fig
F_{LPC}	LPC clock Frequency		25	MHz	1	
T_{Duty}	LPC clock Duty Cycle	45	55	%	1	
T_{CO}	LPC_AD[3:0], LPC_FRAMEB, LPC_CLKRUNB, LPC_SERIRQ Valid Delay w.r.t. rising edge of LPC_CLK	1	12	ns	1,4	91
T_{Float}	LPC_AD[3:0] float delay w.r.t rising edge of LPC_CLK	-	40	ns	1,4,5	93
T_{Setup}	LPC_AD[3:0], LPC_FRAMEB, LPC_CLKRUNB, LPC_SERIRQ Setup Time to rising edge of LPC_CLK	16	-	ns	2,3	94
T_{Hold}	LPC_AD[3:0], LPC_FRAMEB, LPC_CLKRUNB, LPC_SERIRQ hold Time to rising edge of LPC_CLK	0	-	ns	2,3	94
$T_{Rise/fall\ 1.8V}$	Minimum and Maximum Rise/Fall Time at 1.8V		1.5	ns	6	
$T_{Rise/fall\ 3.3V}$	Minimum and Maximum Rise/Fall Time at 3.3V		1.5	ns	7	

NOTE:
1. Timing is measured from 50% to 50% Vdd
2. Minimum setup/hold timing for SoC input for proper functionality across all PVT corners.
3. SoC input timings are measured at SoC pad from 50% to 50%. (Assuming 1ns signal input rise/fall time measured across 20% to 80% of Vcc)
4. SoC output timings are measured at SoC pad with a test load of 12pF
5. When LPC is operating at 19.2 MHz, Tfloat = 52ns
6. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 35%/ 65% of Vdd.
7. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 25%/ 62.5% of Vdd.

Figure 96. Valid Delay from Rising Clock Edge

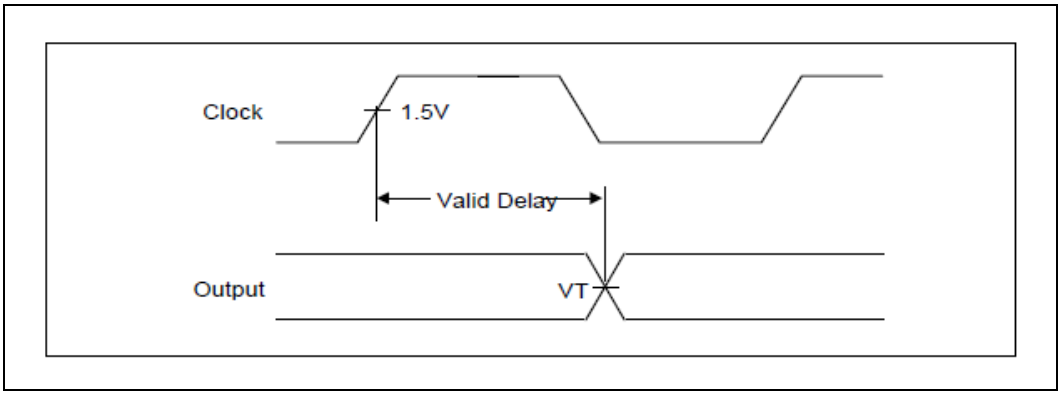


Figure 97. Float Delay

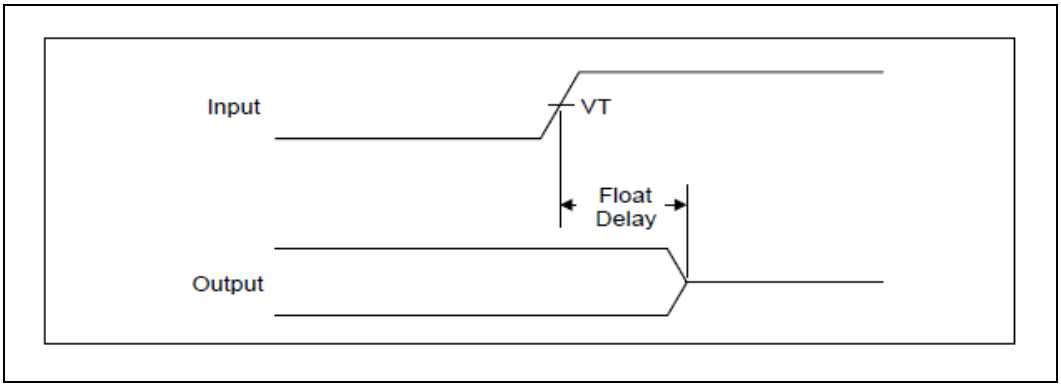
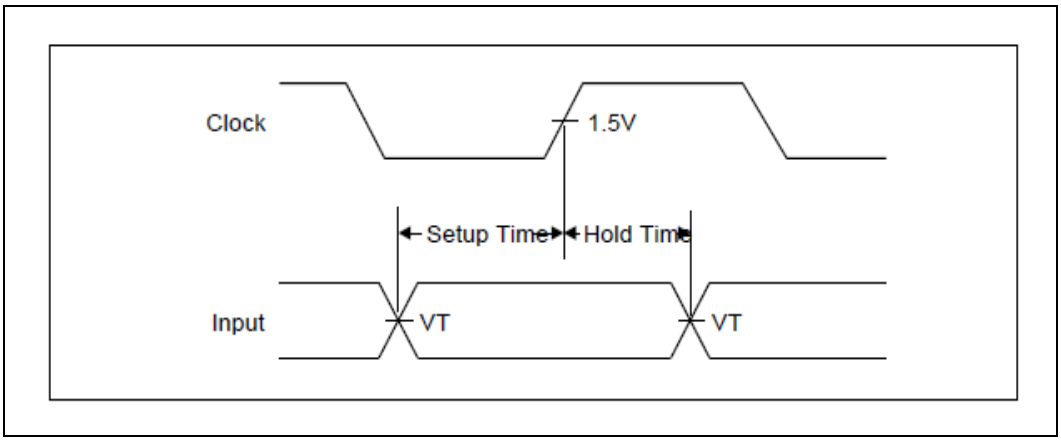


Figure 98. Setup and Hold Times



20.7.19 I²C AC Specification

20.7.19.1 I²C Fast/Standard Mode Electrical Specification

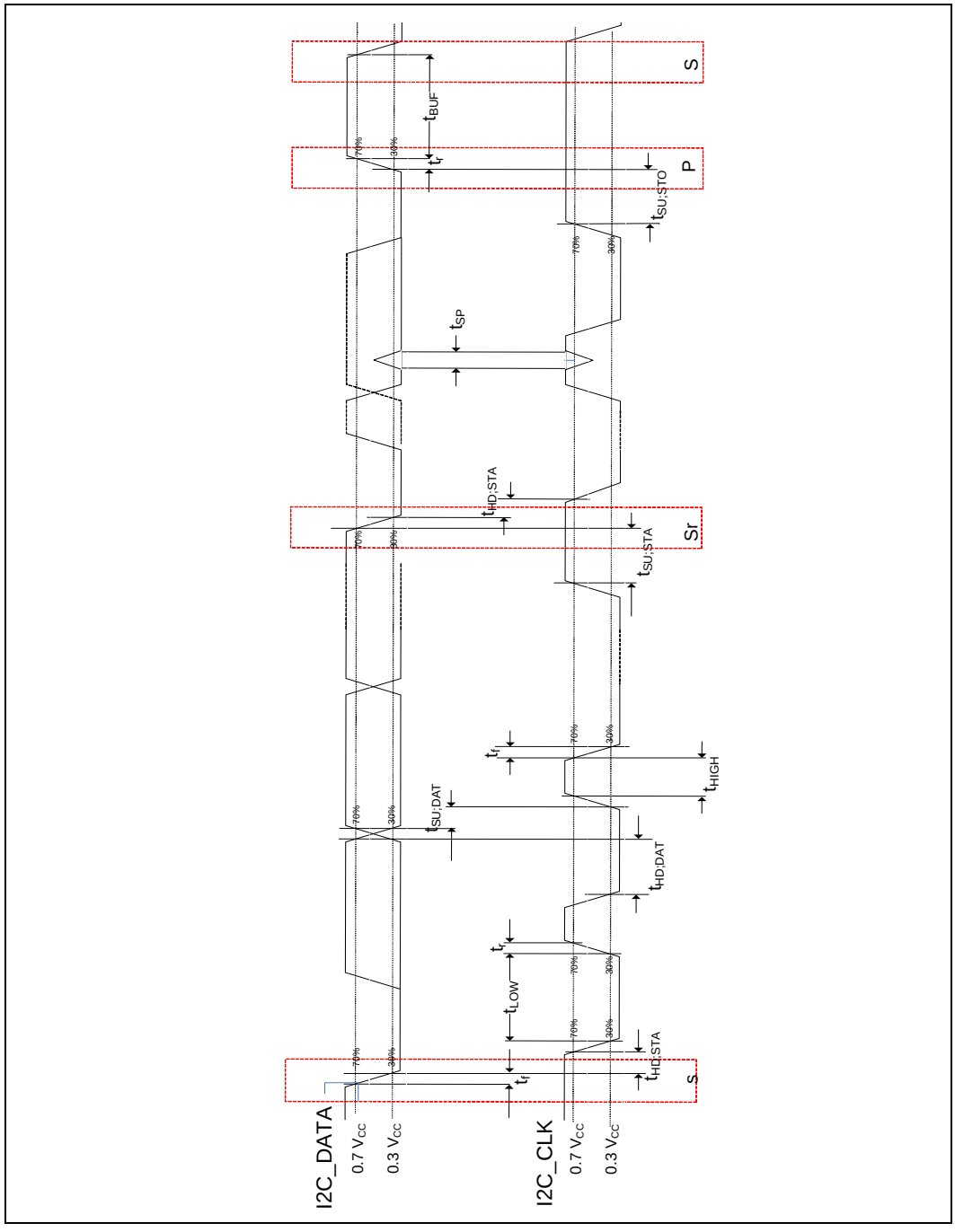
Table 181. I²C Fast/Standard Mode AC Specifications

Symbol	Parameter	Standard-Mode		Fast-Mode		Fast-Mode Plus		Units	Notes	Figure
		Min.	Max.	Min.	Max.	Min.	Max.			
f_{SCL}	I²C_CLK clock frequency	0	100	0	400	0	1000	kHz		
$t_{HD:STA}$	Hold time (repeated) START condition. After this period, the first clock pulse is generated	4.0	–	0.6	–	0.26	–	µs		99
t_{LOW}	LOW period of the I²C_CLK clock	4.7	–	1.3	–	0.5	–	µs		99
t_{HIGH}	HIGH period of the I²C_CLK clock	4.0	–	0.6	–	0.26	–	µs		99
$t_{SU:STA}$	Set-up time for a repeated START condition	4.7	–	0.6	–	0.26	–	µs		99
$t_{HD:DAT}$	Data hold time: I²C-bus devices	0	–	0	–	0	–	ns		99
$t_{SU:DAT}$	Data set-up time	250	–	100	–	50	–	ns	1	99
t_r	Rise time of both I²C_DATA and I²C_CLK signals	–	1000	$20 + 0.1C_b$ (5)	300	–	120	ns	2, 3	99
t_f	Fall time of both I²C_DATA and I²C_CLK signals	1	300	1	300	1	120	ns	5	99
$t_{SU:STO}$	Set-up time for STOP condition	4.0	–	0.6	–	0.26	-	µs		99
t_{BUF}	Bus free time between a STOP and START condition	4.7	–	1.3	–	0.5	–	µs		
C_b	Capacitive load for each bus line	–	130	–	130	-	TBD	pF		
V_{nL}	Noise margin at the LOW level for each connected device (including hysteresis)	$0.1 V_{DD}$	–	$0.1 V_{DD}$	–	$0.1 V_{DD}$	–	V		
V_{nH}	Noise margin at the HIGH level for each connected device (including hysteresis)	$0.2 V_{DD}$	–	$0.2 V_{DD}$	–	$0.2 V_{DD}$	–	V		

NOTES:

1. A Fast-mode I2C-bus device can be used in a Standard-mode I2C-bus system, but the requirement tSU; DAT ³ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the I2C_CLK signal. If such a device does stretch the LOW period of the I2C_CLK signal, it must output the next data bit to the I2C_DATA line tr max + tSU; DAT = 1000 + 250 = 1250 ns (according to the Standard-mode I2C-bus specification) before the I2C_CLK line is released
2. Cb = total capacitance of one bus line in pF.
3. No Active current source PU on I2C_CLK signals. Rise time is based upon the Pull-up resistor mentioned in the Platform Design Guide.
4. The maximum tHD;DAT could be 3.45 ms and 0.9 ms for Standard-mode and Fast-mode, but must be less than the maximum of tVD;DAT or tVD;ACK by a transition time. This maximum must only be met if the device does not stretch the LOW period (tLOW) of the I2C_CLK signal. If the clock stretches the I2C_CLK, the data must be valid by the set-up time before it releases the clock.
5. Specification deviates from the minimum time compared to Industrial specification.

Figure 99. Definition of Timing for F/S-Mode Devices on I²C Bus

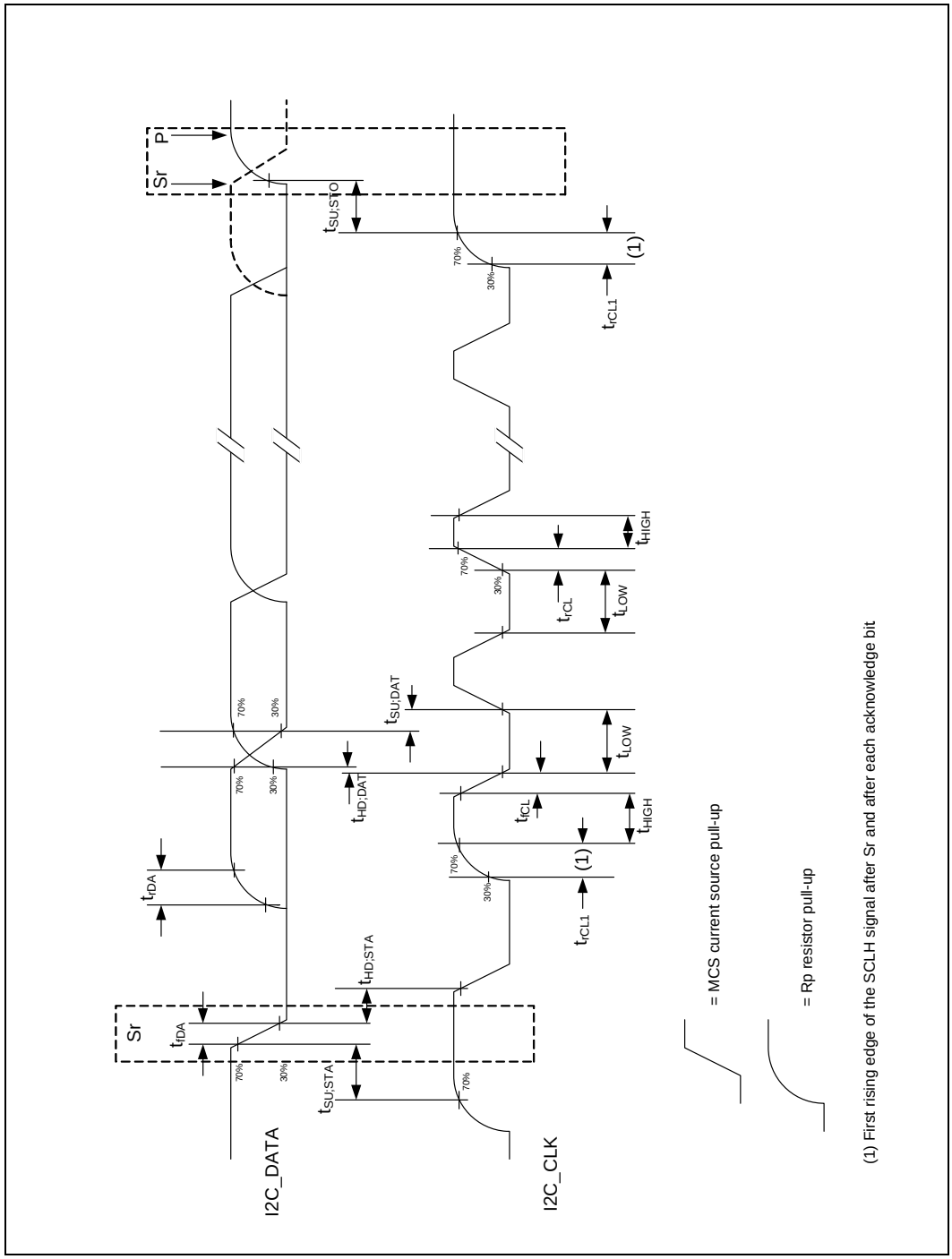


20.7.19.2 I²C High Speed Mode Electrical Specification

Table 182. AC Specification for High Speed Mode I²C—Bus Devices

Symbol	Parameter	C_b = 100 pF (max)		Units	Figure
		Min.	Max.		
f_{SCL}	I²C_CLK clock frequency	0	1.7	MHz	
t_{SU}:STA	Set-Up time for a repeated START condition	160	–	ns	
t_{HD}:STA	Hold time (repeated) START condition.	160	–	ns	
t_{LOW}	LOW period of the I²C_CLK clock	160	–	ns	
t_{HIGH}	HIGH period of the I²C_CLK clock	60	–	ns	
t_{HD}:DAT	Data hold time: I²C-bus devices	0	–	ns	
t_{SU}:DAT	Data set-up time	10	–	ns	
t_r CL	Rise time of I²C_CLK signals	10	40	ns	
t_f CL	Fall time of I²C_CLK signals	1	40	ns	
t_rCL1	Rise time of I²C_CLK signal after a repeated START condition and after an acknowledge bit	10	40	ns	
t_r DA	Rise time of I²C_DATA signals	10	80	ns	
t_f DA	Fall time of I²C_DATA signals	1	80	ns	
$t_{SU:}$STO	Set-up time for STOP condition	160	–	ns	
V_{nL}	Noise margin at the LOW level for each connected device (including hysteresis)	0.1 V_{DD}	–	V	
V_{NH}	Noise margin at the HIGH level for each connected device (including hysteresis)	0.2 V_{DD}	–	V	

Figure 100.Definition of Timing for High Speed-Mode Devices on I²C Bus



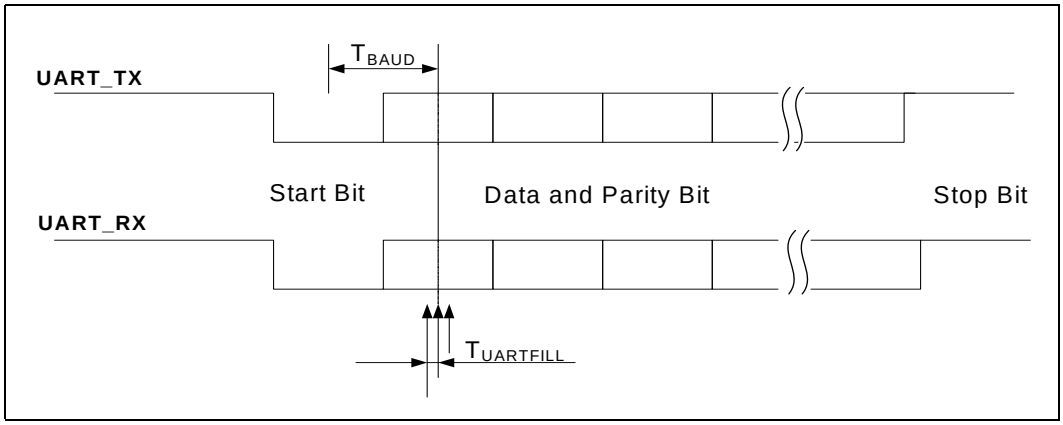
20.7.20 UART AC Specification

Table 183. UART AC Specification

Symbol	Parameter	Min.	Max.	Unit	Notes
F_{UART}	UART sample clock Frequency		1.8432/15	MHz	1
T_{DUTY}	UART sample clock Duty Cycle	45	55	%	1
T_{RISE}	Maximum Rise Time	0.2	1.5	ns	2
T_{FALL}	Maximum Fall Time	0.2	1.5	ns	2
$T_{UARTFIL}$	UART Sampling Filter Period	20		ns	1

NOTES:
1. Timing is measured from 50% to 50% Vdd.
2. SoC output signal rise/fall time spec is measured with test load of 15pF across voltage threshold 35%/65%

Figure 101.UART Timing Diagram



20.7.21 JTAG AC Specification

Table 184. JTAG AC Specification (Sheet 1 of 2)

T# Parameter	Min	Max	Unit	Figure	Notes
T_P:JTAG_TCK Period	15		ns		66 MHz
T_{CL}:JTAG_TCK Clock Low Time	0.4 * T_{JC}		ns		
T_{CH}:JTAG_TCK Clock High Time	0.4 * T_{JC}		ns		
T_{SU}:JTAG_TDI, JTAG_TMS Setup Time	11		ns	103	
T_H: JTAG_TDI, JTAG_TMS Hold Time	5		ns	103	

Table 184. JTAG AC Specification (Sheet 2 of 2)

T# Parameter	Min	Max	Unit	Figure	Notes
T_{CO}: JTAG_TCK falling to JTAG_TDO output valid		11	ns	103	
T_{CO}: JTAG_TCK falling to JTAG_TDO output high impedance		11	ns		
T_{18}: JTAG_TRST# assert time	2		ns	104	

NOTES:
1. Unless otherwise noted, all specifications in this table apply to all SoC frequencies.
2. Not 100% tested. Specified by design characterization.
3. It is recommended that JTAG_TMS be asserted while JTAG_TRST# is being deasserted.
4. Board JTAG signal skew max = ±500 ps.

Figure 102.JTAG Timing Diagram

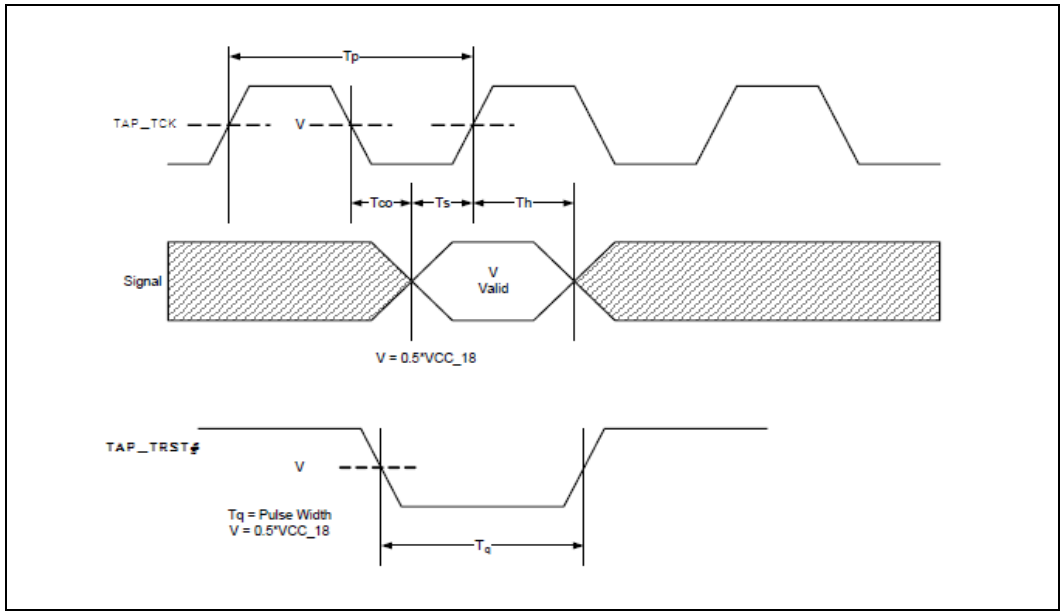
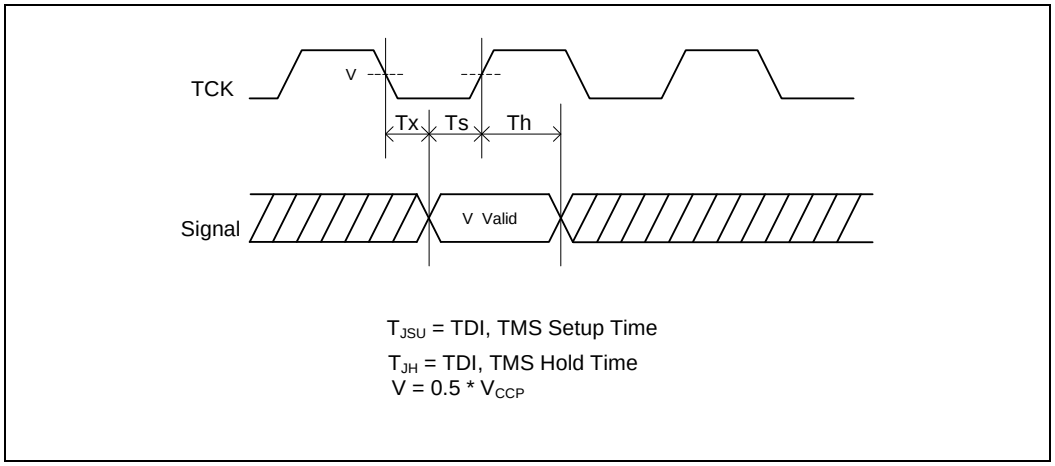


Table 185. Boundary Scan AC Specification

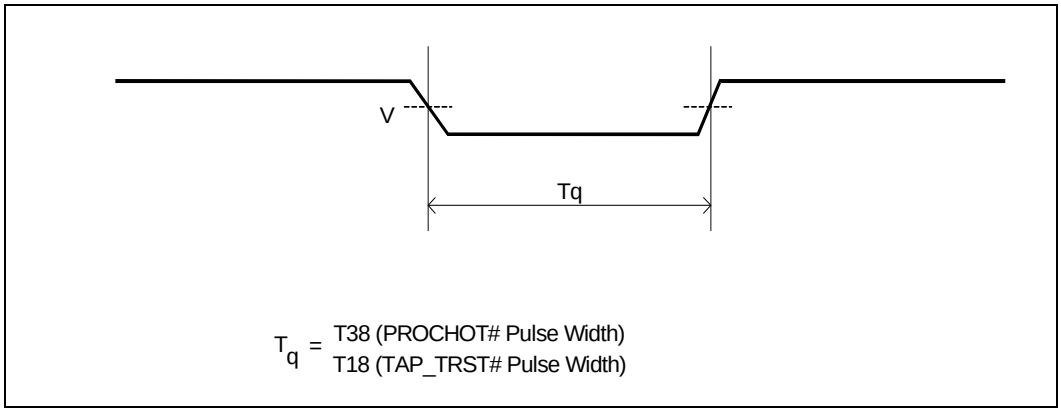
T# Parameter	Min	Max	Unit	Notes
Boundary scan all non test output/float delay	0.5	15	ns	Referenced to the falling edge of T_{CK}
Boundary scan all non test input setup	10		ns	Referenced to the falling edge of T_{CK}
Boundary scan all non test input hold	13		µs	Referenced to the falling edge of T_{CK}

Figure 103.JTAG Valid Delay Timing Waveform



Refer to Table 133, Table 134, Table 135 for JTAG Signal Group DC specifications and Table 184 for JTAG Signal Group AC specifications.

Figure 104.Test Reset (JTAG_TRST#), Async GTL Input and PROCHOT# Timing Waveform



20.7.22 PCI Express AC Specification

Table 186. PCI Express Interface Timings

Symbol	Parameter	Min	Typ	Max	Unit	Figures	Notes
Transmitter and Receiver Timings							
UI	Unit Interval – PCI Express* Gen 1 (2.5 GT/s)	399.88		400.12	ps		5
UI	Unit Interval – PCI Express* Gen 2 (5.0 GT/s)	199.9		200.1	ps		5
T_{TX-EYE}	Minimum Transmission Eye Width	0.7		—	UI	112	1,2

Table 186. PCI Express Interface Timings

$T_{TX-RISE/Fall}$ (Gen1)	TXP/TXN Rise/Fall time	0.125			UI		1,2
$T_{TX-RISE/Fall}$ (Gen2)	TXP/TXN Rise/Fall time	0.15			UI		1,2
T_{RX-EYE}	Minimum Receiver Eye Width	0.40		—	UI	113	3,4

NOTES:

1. Specified at the measurement point into a timing and voltage compliance test load and measured over any 250 consecutive TX UIs. (Also refer to the Transmitter compliance eye diagram)
2. A T_{TX-EYE} = 0.70 UI provides for a total sum of deterministic and random jitter budget of $T_{TXJITTER-MAX}$ = 0.30 UI for the Transmitter collected over any 250 consecutive TX UIs. The $T_{TXEYE-MEDIAN-to-MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total TX jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value.
3. Specified at the measurement point and measured over any 250 consecutive UIs. The test load documented in the PCI Express* specification 2.0 should be used as the RX device when taking measurements (also refer to the Receiver compliance eye diagram). If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as a reference for the eye diagram.
4. A T_{RX-EYE} = 0.40 UI provides for a total sum of 0.60 UI deterministic and random jitter budget for the Transmitter and interconnect collected any 250 consecutive UIs. The $T_{RX-EYE-MEDIAN-to--MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total 0.6 UI jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value. If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as the reference for the eye diagram.
5. Nominal Unit Interval is 400 ps for 2.5 GT/s and 200 ps for 5 GT/s.
6. PCIe Reference clocks follow PCI Express* specification with the exception of edge rate: Max = 8.0 V/ns instead of 4.0 V/ns. There should be no DC termination of the clocks.

Figure 105.PCI Express* Transmitter Eye

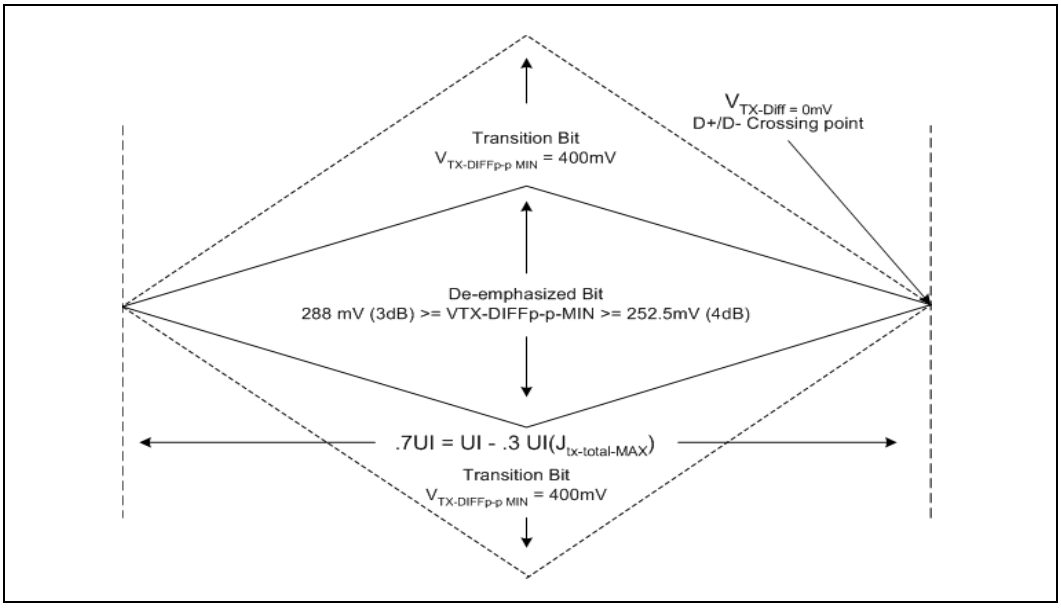
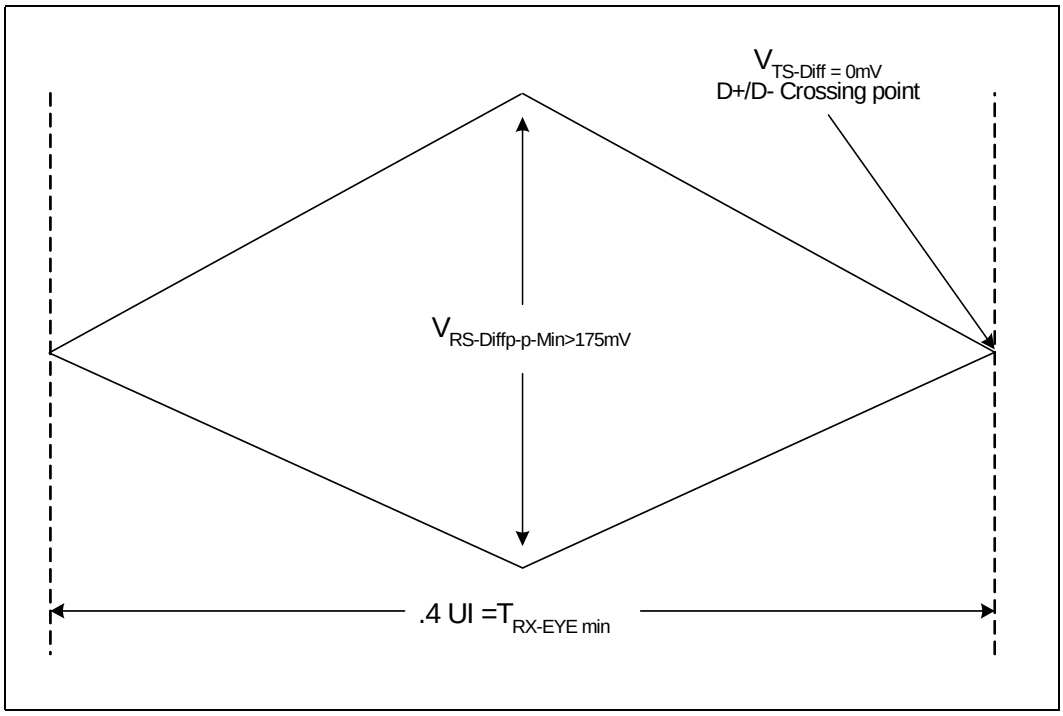


Figure 106.PCI Express* Receiver Eye



20.7.23 General AC Timing Diagrams

Note that the measurement of the differential waveform according to these diagrams would have to be made directly at the load at the end of the line. In a real system, this is not possible because the end of the line is at the input pad of the SoC silicon.

Figure 107.Clock Cycle Time

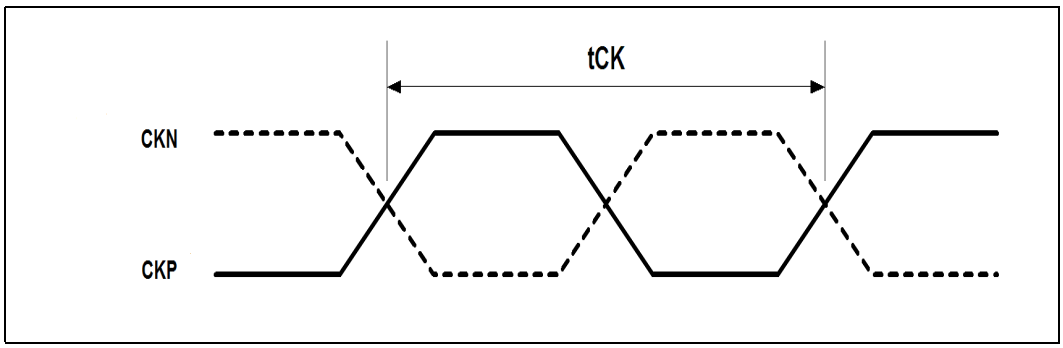


Figure 108.Clock Timing

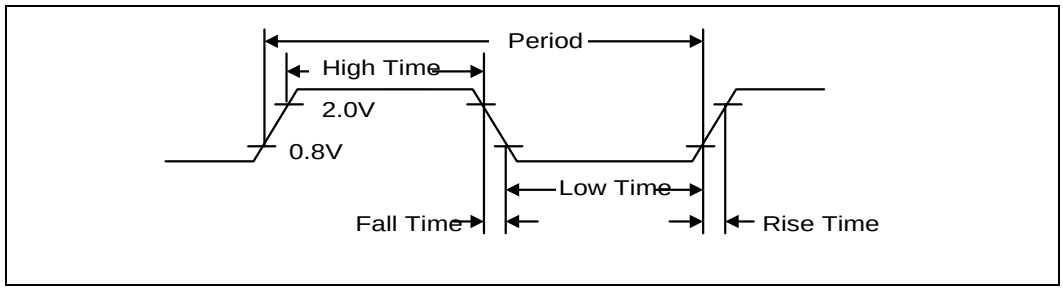


Figure 109.Valid Delay from Rising Clock Edge

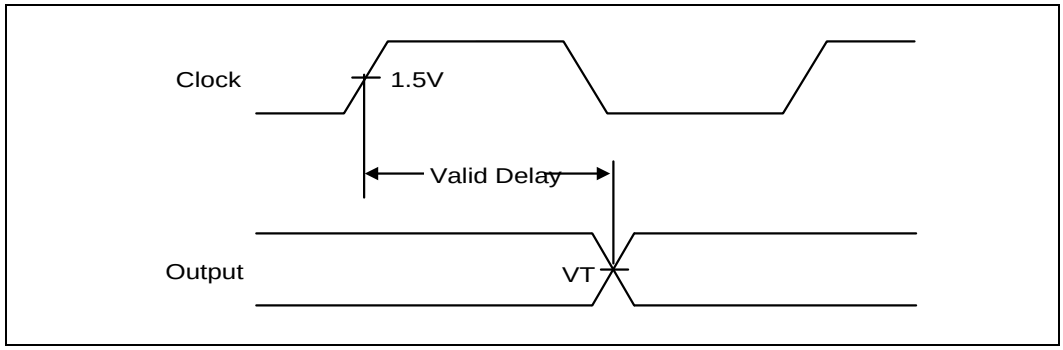


Figure 110.Setup and Hold Times

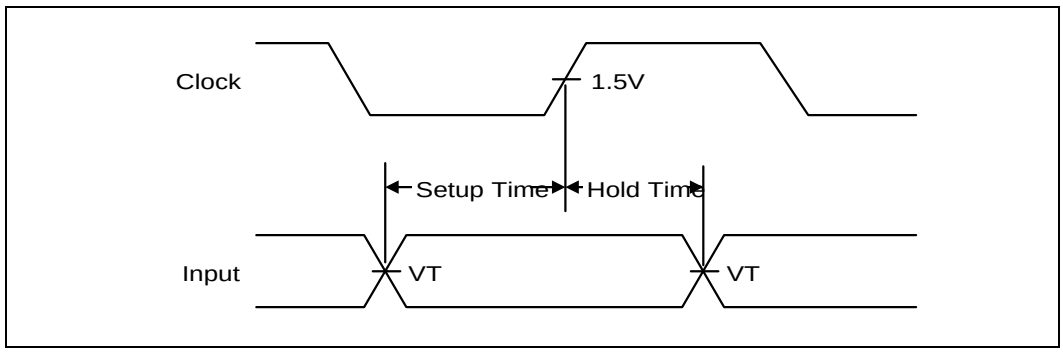


Figure 111.Float Delay

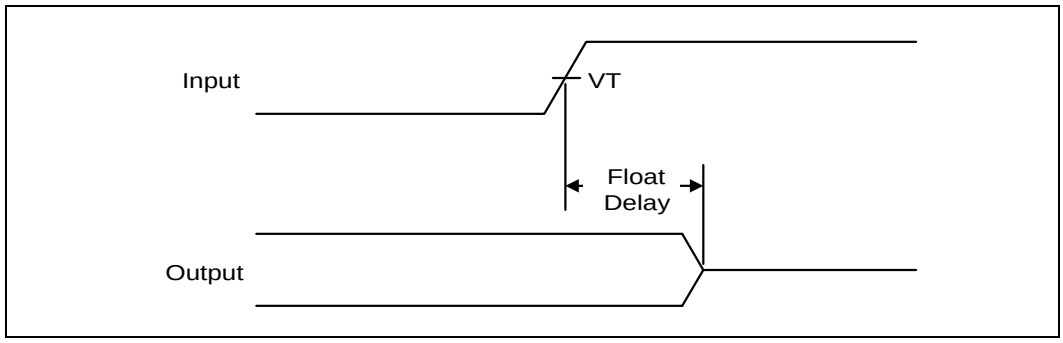


Figure 112.Pulse Width

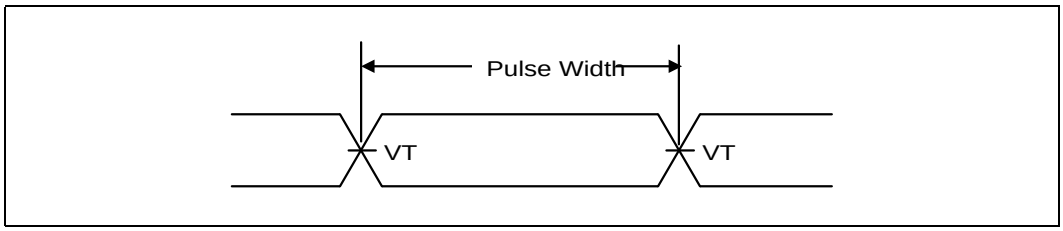


Figure 113.Output Enable Delay

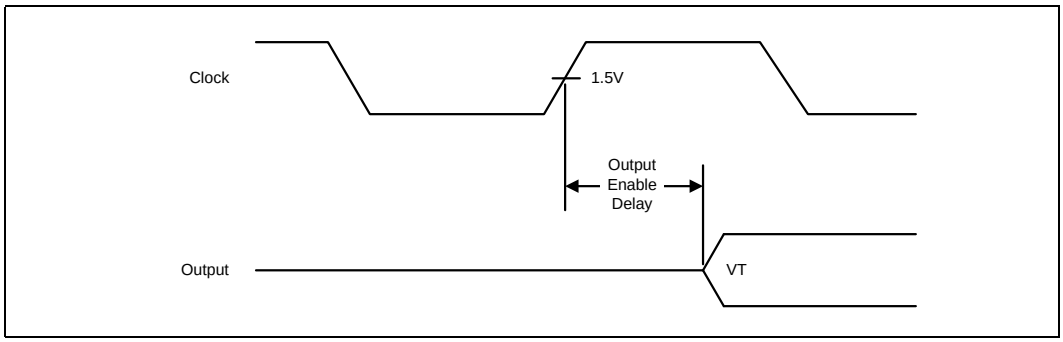


Figure 114.Differential Clock Waveform (Measured Single-ended)

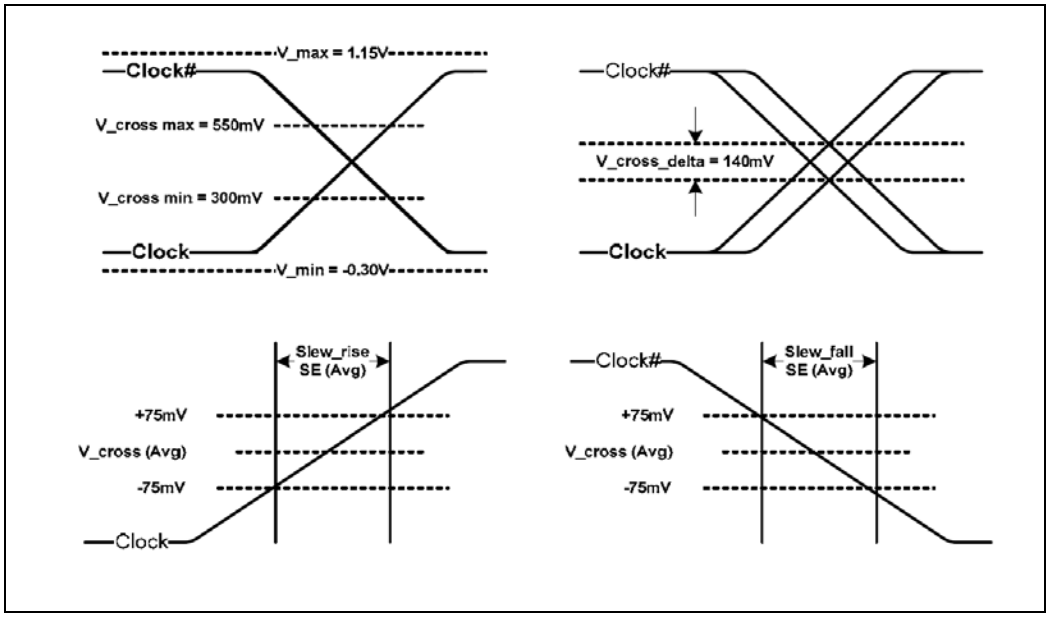
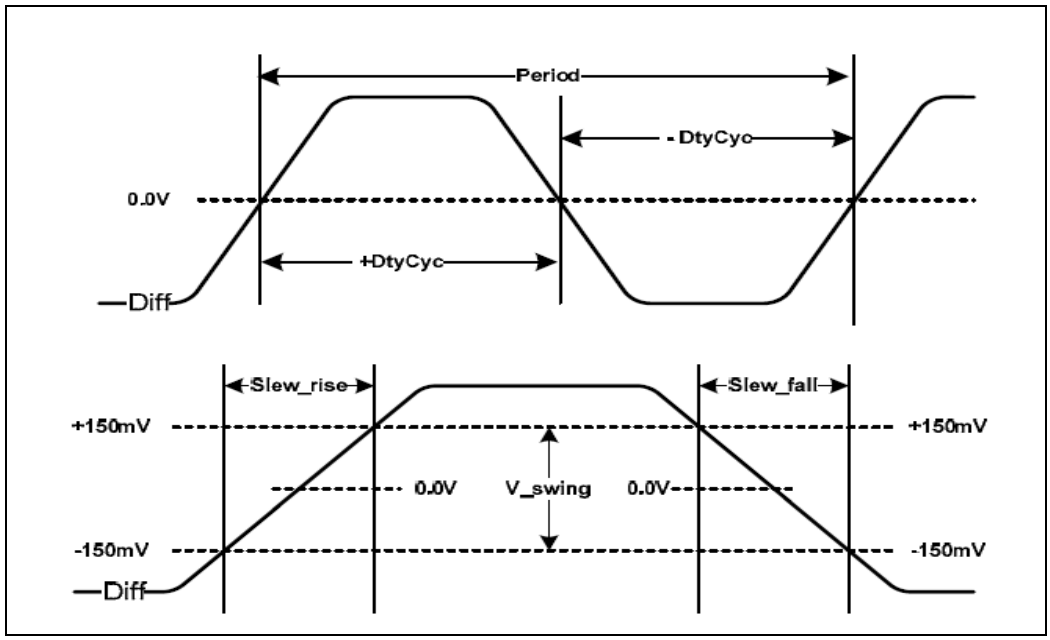
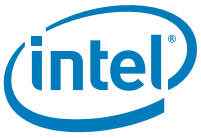


Figure 115.Differential Clock Waveform (Using Differential Probe for Measurement)



§



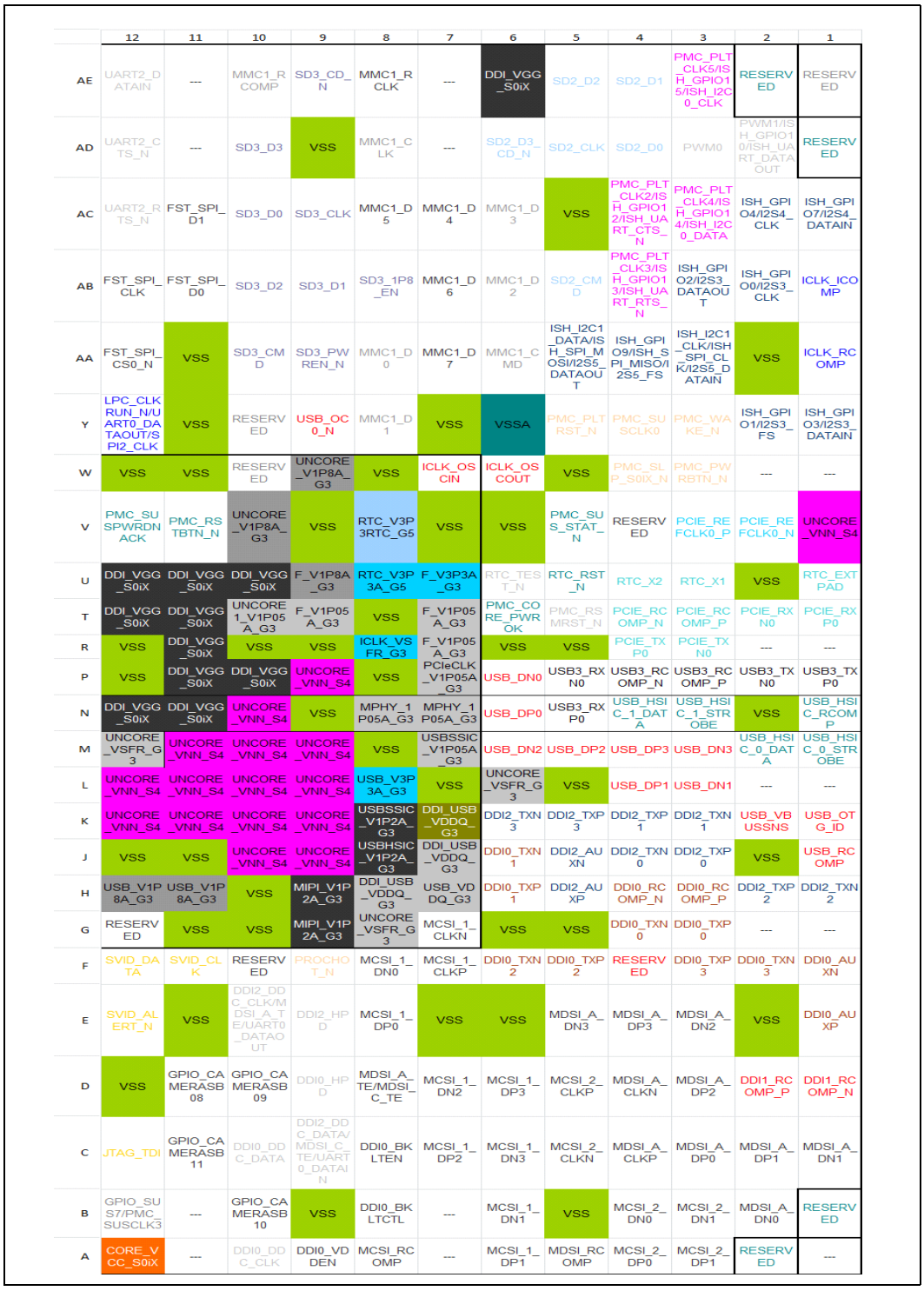
21 Ballout and Ball Map

21.1 Ballout

Figure 116.Ballout - DDR3L-RS (VMS-T3) Top View Part A

	25	24	23	22	21	20	19	18	17	16	15	14	13
AE	PWR_RSVD_OBS	PWR_RSVD_OBS	DDR3_M0_DQ35	DDR3_M0_ODQVREF	DDR3_M0_OCAVREF	DDR3_M0_RCOMPPD	---	UART0_DATAIN	I2C5_DATA	I2C6_CLK/NMI_N	---	SD3_RCOMP	UART1_RTS_N
AD	VSS	DDR3_M0_DQ32	DDR3_M0_DQ38	VSS	DDR3_M0_DQ51	/SD3_WP	---	MMC1_RESET_N	VSS	I2C6_DATA/SD3_WP	---	UART1_DATAIN/UART0_DATAIN	VSS
AC	DDR3_M0_DQS4_P	DDR3_M0_DM4	DDR3_M0_DQ37	DDR3_M0_DQ49	RESERVED	GPIO_SW93	LPE_I2S2_FRM	GPIO_SW78	I2C2_DATA	NFC_I2C_CLK	NFC_I2C_DATA	UART1_DATAOUT/UART0_DATAOUT	UART1_CTS_N
AB	DDR3_M0_DQ33	DDR3_M0_DQS4_N	DDR3_M0_DQS6_P	DDR3_M0_DQ48	RESERVED	PCIE_CLKREQ0_N/GPIO_SW90	LPE_I2S2_CLK	I2C4_DATA/DDI2_DDC_DATA	I2C4_CLK/DDI2_DDC_CLK	I2C5_CLK	LPE_I2S0_DATAIN	LPE_I2S1_CLK	UART2_DATAOUT
AA	DDR3_M0_DQ36	VSS	DDR3_M0_DQS6_N	DDR3_M0_DQ52	VSS	DDR3_CORE_PWROK	DDR3_DRAM_PWROK	VSS	I2C2_CLK	LPE_I2S0_CLK	LPE_I2S0_DATAOUT	LPE_I2S1_FRM	LPC_SERIRQ/SPI2_CS0_N
Y	DDR3_M0_DQ39	DDR3_M0_DQ34	DDR3_M0_DQ53	DDR3_M0_DM6	DDR3_M0_DQ55	DDR3_M0_DQ54	DDR3_M0_DQ50	I2C0_DATA	I2C0_CLK	LPE_I2S0_FRM	DDI_VGG_S0iX	LPE_I2S1_DATAOUT	LPE_I2S1_DATAIN
W	---	---	DDR3_M0_DQ43	DDR3_M0_DQ42	DDR3_M0_DQ61	DDR3_M0_DQ58	VSS	VSS	VSS	I2C1_CLK	I2C1_DATA	DDI_VGG_S0iX	SDIO_V3P3A_V1P8A_G3
V	DDR3_M0_DQS5_P	DDR3_M0_DM5	DDR3_M0_DQ40	DDR3_M0_DQ46	VSS	DDR3_M0_DQ59	DDR_VDDQG_S4	LPE_I2S2_DATAOUT	DDI_VGG_S0iX	UNCORE_V1P8A_G3	UNCORE_V1P8A_G3	DDI_VGG_S0iX	LPC_FRAME_N/UART0_DATAIN/SPI2_MISO
U	DDR3_M0_DQS5_N	VSS	DDR3_M0_DQS7_N	DDR3_M0_DQS7_P	DDR3_M0_DQ60	DDR3_M0_DQ62	DDR3_M0_DRAMRST_N	DDR_V1P05A_G3	LPE_I2S2_DATAIN	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX
T	DDR3_M0_DQ47	DDR3_M0_DQ41	DDR3_M0_DM7	DDR3_M0_DQ56	DDR3_M0_DQ63	DDR3_M0_DQ57	DDR_VDDQG_S4	VSS	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX
R	---	---	DDR3_M0_DQ44	DDR3_M0_DQ45	VSS	VSS	VSS	DDR_V1P05A_G3	DDI_V1p05A_S0iX	DDI_V1p05A_S0iX	VSS	DDI_VGG_S0iX	DDI_VGG_S0iX
P	DDR3_M0_ODT1	DDR3_M0_ODT0	DDR3_M0_CS1_N	DDR3_M0_CS0_N	DDR3_M0_WE_N	DDR3_M0_CAS_N	DDRSFR_VDDQG_S4	VSS	VSS	VSS	VSS	VSS	RESERVED
N	DDR3_M0_MA7	VSS	DDR3_M0_MA2	DDR3_M0_RAS_N	VSS	DDR3_M0_MA10	DDR3_M0_BS1	VSS	VSS	VSS	CORE_VCC_S0iX	CORE_VCC_S0iX	VSS
M	DDR3_M0_MA12	DDR3_M0_MA0	DDR3_M0_MA13	DDR3_M0_MA4	DDR3_M0_BS0	DDR3_M0_MA3	VSS	DDR_VDDQG_S4	VSS	VSS	CORE_VCC_S0iX	CORE_VCC_S0iX	VSS
L	---	---	DDR3_M0_MA1	DDR3_M0_MA5	VSS	DDR_VDDQG_S4	DDR_VDDQG_S4	DDR_VDDQG_S4	CORE_V1p05A_S0iX	F_V1p05A_S0iX	CORE_VCC_S0iX	CORE_VCC_S0iX	UNCORE2_V1P05A_G3
K	DDR3_M0_MA11	DDR3_M0_MA6	DDR3_M0_MA15	DDR3_M0_BS2	DDR3_M0_CK1_N	DDR3_M0_CK1_P	VSS	DDR_VDDQG_S4	CORE_V1p05A_S0iX	F_V1p05A_S0iX	CORE_VCC_S0iX	VSS	VSS
J	DDR3_M0_MA8	VSS	DDR3_M0_CK0_N	DDR3_M0_CK0_P	DDR3_M0_DQ29	DDR3_M0_DQ30	VSS	VSS	VSS	VSS	CORE_VCC_S0iX	CORE_VCC_S0iX	VSS
H	DDR3_M0_MA14	DDR3_M0_MA09	DDR3_M0_DQ27	DDR3_M0_DQ31	VSS	DDR3_M0_DQS3_P	DDR3_M0_DQS3_N	DDR_V1P05A_G3	VSS	VSS	CORE_VCC_S0iX	CORE0_VSFR_G3	UNCORE_V1p05A_S0iX
G	---	---	DDR3_M0_CKE1	DDR3_M0_CKE0	DDR3_M0_DQ28	DDR3_M0_DQ26	DDR_V1P05A_G3	CORE_V1p05A_S0iX	CORE_V1p05A_S0iX	VSS	CORE_VCC_S0iX	CORE_VCC_S0iX	UNCORE_V1p05A_S0iX
F	DDR3_M0_DQ19	DDR3_M0_DQ23	DDR3_M0_CKE2	DDR3_M0_CKE3	DDR3_M0_DQ25	DDR3_M0_DQ24	DDR3_M0_DM3	GPIO_DFX4	GPIO_DFX3/C0_BPM3_TX	GPIO_DFX7/C0_BPM2_TX	VSS	JTAG_TMS	JTAG_TDO
E	DDR3_M0_DQ16	VSS	DDR3_M0_DQ11	DDR3_M0_DQ13	VSS	DDR3_M0_DQ14	DDR3_M0_DQ10	VSS	GPIO_DFX0/C0_BPM0_TX	GPIO_DFX5/C0_BPM0_TX/C1_BPM0_TX	VSS	JTAG_TCK	JTAG_TRST_N
D	DDR3_M0_DM2	DDR3_M0_DQ21	DDR3_M0_DQ9	DDR3_M0_DQ12	DDR3_M0_DM1	DDR3_M0_DQ15	DDR3_M0_DQS0_P	DDR3_M0_DQ0	GPIO_DFX2/ISH_GPIO13/C0_BPM2_TX	GPIO_DFX6/C0_BPM1_TX/C1_BPM1_TX	GPIO_SUS2/JTAG2_TMS	GPIO_SUS4/JTAG2_TDO	JTAG_PRDY_N
C	DDR3_M0_DQS2_N	DDR3_M0_DQS2_P	DDR3_M0_DQ18	DDR3_M0_DQS1_P	DDR3_M0_DQS1_N	DDR3_M0_DQ8	DDR3_M0_DQS0_N	DDR3_M0_DQ3	GPIO_DFX1/C0_BPM1_TX/C1_BPM1_TX	GPIO_DFX8/C0_BPM3_TX/C1_BPM3_TX	GPIO_SUS1/JTAG2_TCK	GPIO_SUS6/PMC_SUSCLK2	GPIO_SUS8
B	PWR_RSVD_OBS	DDR3_M0_DQ20	DDR3_M0_DQ22	DDR3_M0_DQ7	VSS	DDR3_M0_DQ5	---	DDR3_M0_DQ1	VSS	GPIO0_RCOMP	---	GPIO_SUS3/JTAG2_TDI	VSS
A	PWR_RSVD_OBS	VSS	DDR3_M0_DQ17	DDR3_M0_DQ4	DDR3_M0_DQ2	DDR3_M0_DM0	---	DDR3_M0_DQ6	DDR_VDDQG_S4	CORE_V1p05A_S0iX	---	GPIO_SUS0	GPIO_SUS9

Figure 117.Ballout - DDR3L-RS (VMS-T3) Top View Part B



	12	11	10	9	8	7	6	5	4	3	2	1
AE	UART2_DATAIN	---	MMC1_RCOMP	SD3_CD_N	MMC1_RCLK	---	DDI_VGG_S0iX	SD2_D2	SD2_D1	PMC_PLT_CLK5/ISH_GPIO15/ISH_I2C0_CLK	RESERVED	RESERVED
AD	UART2_CTS_N	---	SD3_D3	VSS	MMC1_CLK	---	SD2_D3_CD_N	SD2_CLK	SD2_D0	PWM0	PWM1/ISH_GPIO10/ISH_UART_DATAOUT	RESERVED
AC	UART2_RTS_N	FST_SPI_D1	SD3_D0	SD3_CLK	MMC1_D5	MMC1_D4	MMC1_D3	VSS	PMC_PLT_CLK2/ISH_GPIO12/ISH_UART_CTS_N	PMC_PLT_CLK4/ISH_GPIO14/ISH_I2C0_DATA	ISH_GPIO4/I2S4_CLK	ISH_GPIO7/I2S4_DATAIN
AB	FST_SPI_CLK	FST_SPI_D0	SD3_D2	SD3_D1	SD3_1P8_EN	MMC1_D6	MMC1_D2	SD2_CMD	PMC_PLT_CLK3/ISH_GPIO13/ISH_UART_RTS_N	ISH_GPIO2/I2S3_DATAOUT	ISH_GPIO0/I2S3_CLK	ICLK_ICOMP
AA	FST_SPI_CS0_N	VSS	SD3_CMD	SD3_PWREN_N	MMC1_D0	MMC1_D7	MMC1_CMD	ISH_I2C1_DATA/ISH_SPI_MOSI/I2S5_DATAOUT	ISH_GPIO9/ISH_SPI_MISO/I2S5_FS	ISH_I2C1_CLK/ISH_SPI_CLK/I2S5_DATAIN	VSS	ICLK_RCOMP
Y	LPC_CLKRUN_N/UART0_DATAOUT/SPI2_CLK	VSS	RESERVED	USB_OC0_N	MMC1_D1	VSS	VSSA	PMC_PLT_RST_N	PMC_SUSCLK0	PMC_WAKE_N	ISH_GPIO1/I2S3_FS	ISH_GPIO3/I2S3_DATAIN
W	VSS	VSS	RESERVED	UNCORE_V1P8A_G3	VSS	ICLK_OSCIN	ICLK_OSCOUT	VSS	PMC_SLP_S0iX_N	PMC_PWRBTN_N	---	---
V	PMC_SUSPWRDNACK	PMC_RSTBTN_N	UNCORE_V1P8A_G3	VSS	RTC_V3P3RTC_G5	VSS	VSS	PMC_SUS_STAT_N	RESERVED	PCIE_REFCLK0_P	PCIE_REFCLK0_N	UNCORE_VNN_S4
U	DDI_VGG_S0iX	DDI_VGG_S0iX	DDI_VGG_S0iX	F_V1P8A_G3	RTC_V3P3A_G5	F_V3P3A_G3	RTC_TEST_N	RTC_RST_N	RTC_X2	RTC_X1	VSS	RTC_EXTPAD
T	DDI_VGG_S0iX	DDI_VGG_S0iX	UNCORE1_V1P05A_G3	F_V1P05A_G3	VSS	F_V1P05A_G3	PMC_CORE_PWROK	PMC_RSMRST_N	PCIE_RCOMP_N	PCIE_RCOMP_P	PCIE_RXN0	PCIE_RXP0
R	VSS	DDI_VGG_S0iX	VSS	VSS	ICLK_VSFR_G3	F_V1P05A_G3	VSS	VSS	PCIE_TXP0	PCIE_TXN0	---	---
P	VSS	DDI_VGG_S0iX	DDI_VGG_S0iX	UNCORE_VNN_S4	VSS	PCIeCLK_V1P05A_G3	USB_DN0	USB3_RXN0	USB3_RCOMP_N	USB3_RCOMP_P	USB3_TXN0	USB3_TXP0
N	DDI_VGG_S0iX	DDI_VGG_S0iX	UNCORE_VNN_S4	VSS	MPHY_1P05A_G3	MPHY_1P05A_G3	USB_DP0	USB3_RXP0	USB_HSIC_1_DATA	USB_HSIC_1_STROBE	VSS	USB_HSIC_RCOMP
M	UNCORE_VSFR_G3	UNCORE_VNN_S4	UNCORE_VNN_S4	UNCORE_VNN_S4	VSS	USBSSIC_V1P05A_G3	USB_DN2	USB_DP2	USB_DP3	USB_DN3	USB_HSIC_0_DATA	USB_HSIC_0_STROBE
L	UNCORE_VNN_S4	UNCORE_VNN_S4	UNCORE_VNN_S4	UNCORE_VNN_S4	USB_V3P3A_G3	VSS	UNCORE_VSFR_G3	VSS	USB_DP1	USB_DN1	---	---
K	UNCORE_VNN_S4	UNCORE_VNN_S4	UNCORE_VNN_S4	UNCORE_VNN_S4	USBSSIC_V1P2A_G3	DDI_USB_VDDQ_G3	DDI2_TXN3	DDI2_TXP3	DDI2_TXP1	DDI2_TXN1	USB_VBUSSNS	USB_OTG_ID
J	VSS	VSS	UNCORE_VNN_S4	UNCORE_VNN_S4	USBHSIC_V1P2A_G3	DDI_USB_VDDQ_G3	DDI0_TXN1	DDI2_AUXN	DDI2_TXN0	DDI2_TXP0	VSS	USB_RCOMP
H	USB_V1P8A_G3	USB_V1P8A_G3	VSS	MIPI_V1P2A_G3	DDI_USB_VDDQ_G3	USB_VDDQ_G3	DDI0_TXP1	DDI2_AUXP	DDI0_RCOMP_N	DDI0_RCOMP_P	DDI2_TXP2	DDI2_TXN2
G	RESERVED	VSS	VSS	MIPI_V1P2A_G3	UNCORE_VSFR_G3	MCSI_1_CLKN	VSS	VSS	DDI0_TXN0	DDI0_TXP0	---	---
F	SVID_DATA	SVID_CLK	RESERVED	PROCHOT_N	MCSI_1_DN0	MCSI_1_CLKP	DDI0_TXN2	DDI0_TXP2	RESERVED	DDI0_TXP3	DDI0_TXN3	DDI0_AUXN
E	SVID_ALERT_N	VSS	DDI2_DDC_CLK/MDSI_A_TE/UART0_DATAOUT	DDI2_HPD	MCSI_1_DP0	VSS	VSS	MDSI_A_DN3	MDSI_A_DP3	MDSI_A_DN2	VSS	DDI0_AUXP
D	VSS	GPIO_CAMERASB08	GPIO_CAMERASB09	DDI0_HPD	MDSI_A_TE/MDSI_C_TE	MCSI_1_DN2	MCSI_1_DP3	MCSI_2_CLKP	MDSI_A_CLKN	MDSI_A_DP2	DDI1_RCOMP_P	DDI1_RCOMP_N
C	JTAG_TDI	GPIO_CAMERASB11	DDI0_DDC_DATA	DDI2_DDC_DATA/MDSI_C_TE/UART0_DATAIN	DDI0_BKLTEN	MCSI_1_DP2	MCSI_1_DN3	MCSI_2_CLKN	MDSI_A_CLKP	MDSI_A_DP0	MDSI_A_DP1	MDSI_A_DN1
B	GPIO_SUS7/PMC_SUSCLK3	---	GPIO_CAMERASB10	VSS	DDI0_BKLTCTL	---	MCSI_1_DN1	VSS	MCSI_2_DN0	MCSI_2_DN1	MDSI_A_DN0	RESERVED
A	CORE_VCC_S0iX	---	DDI0_DDC_CLK	DDI0_VDDEN	MCSI_RCOMP	---	MCSI_1_DP1	MDSI_RCOMP	MCSI_2_DP0	MCSI_2_DP1	RESERVED	---

Figure 118.Ballout LPPDR3 (MSP-T4) Top View Part A

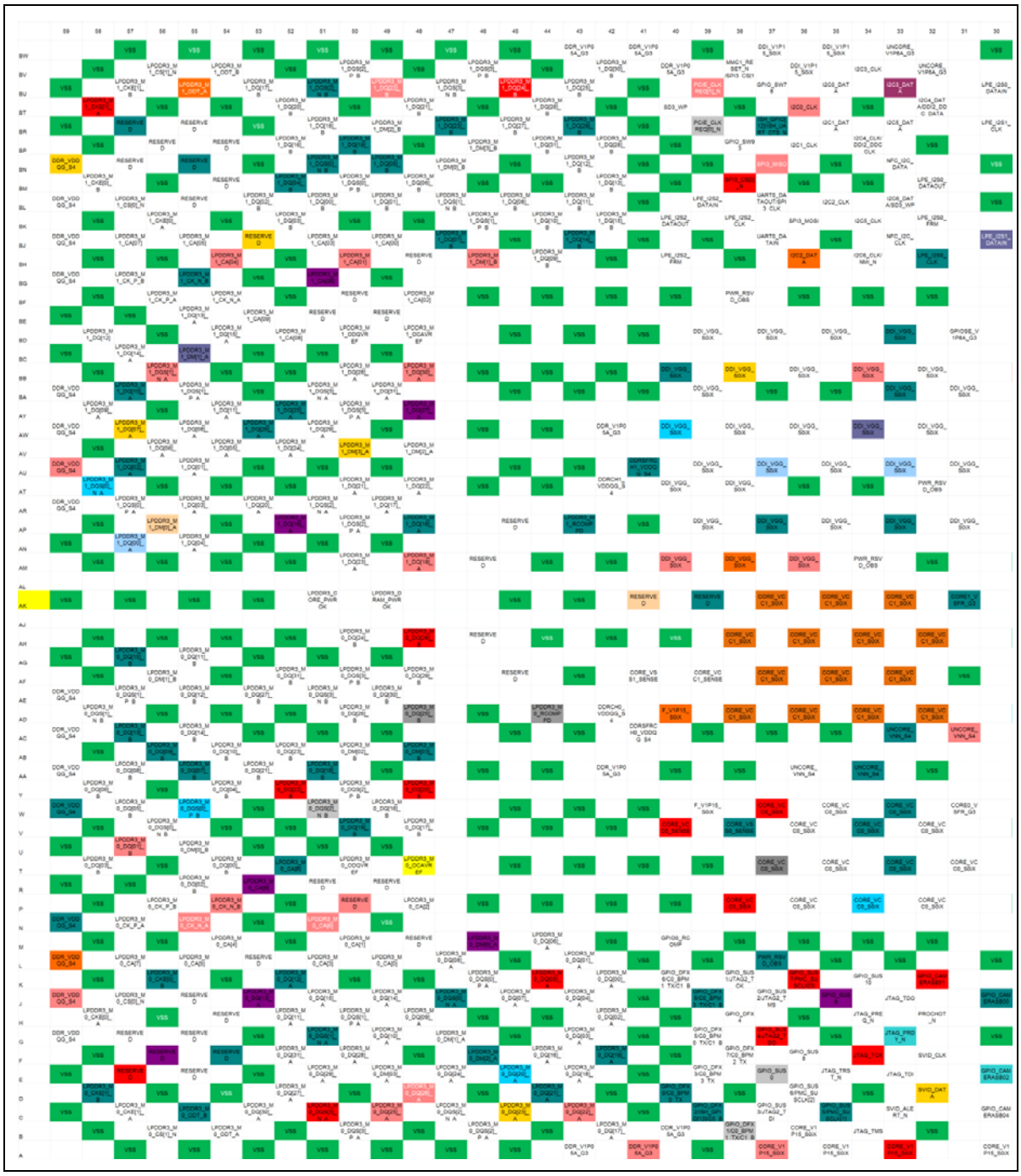


Figure 119.Ballout LPPDR3 (MSP-T4) Top View Part B

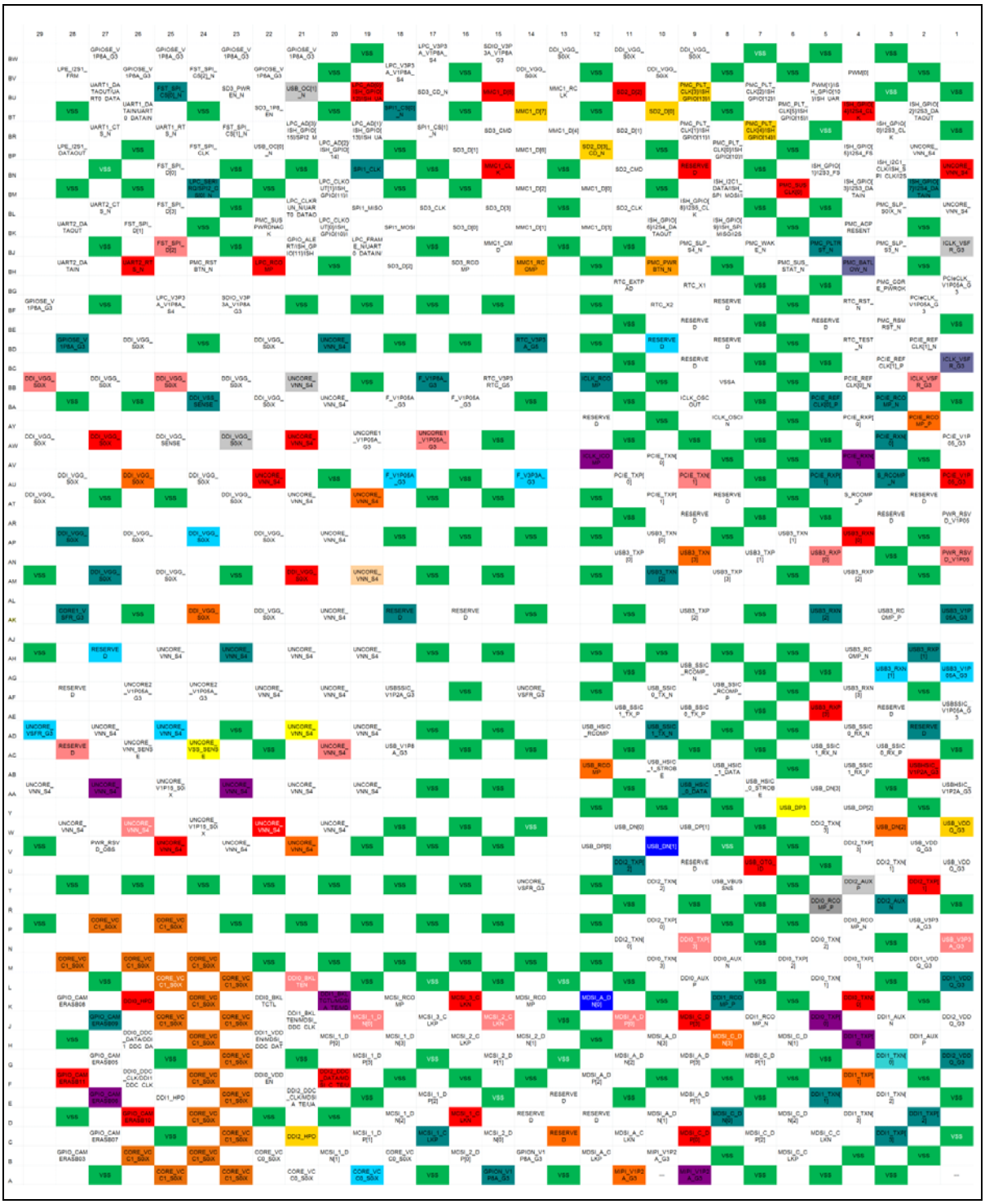
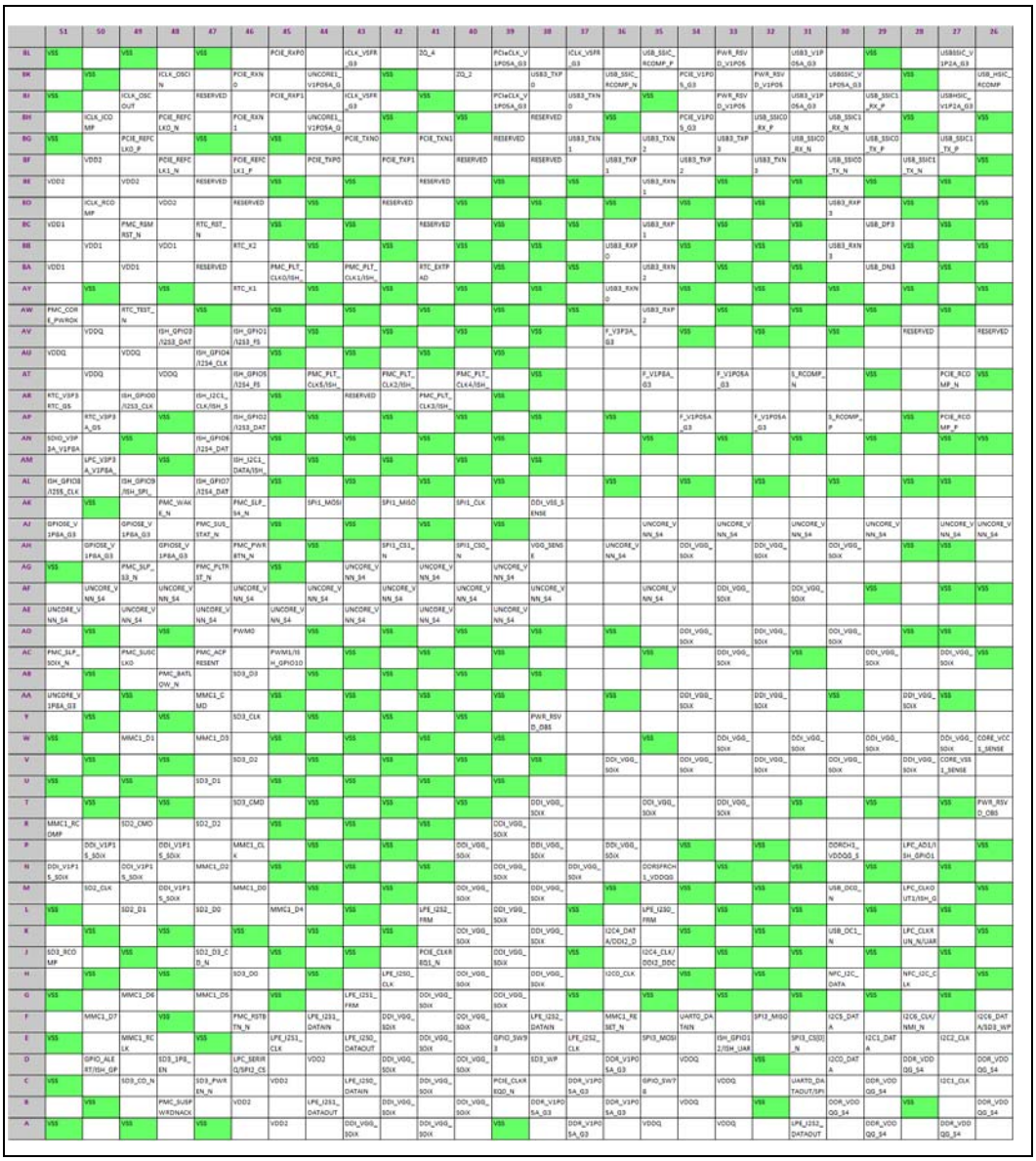
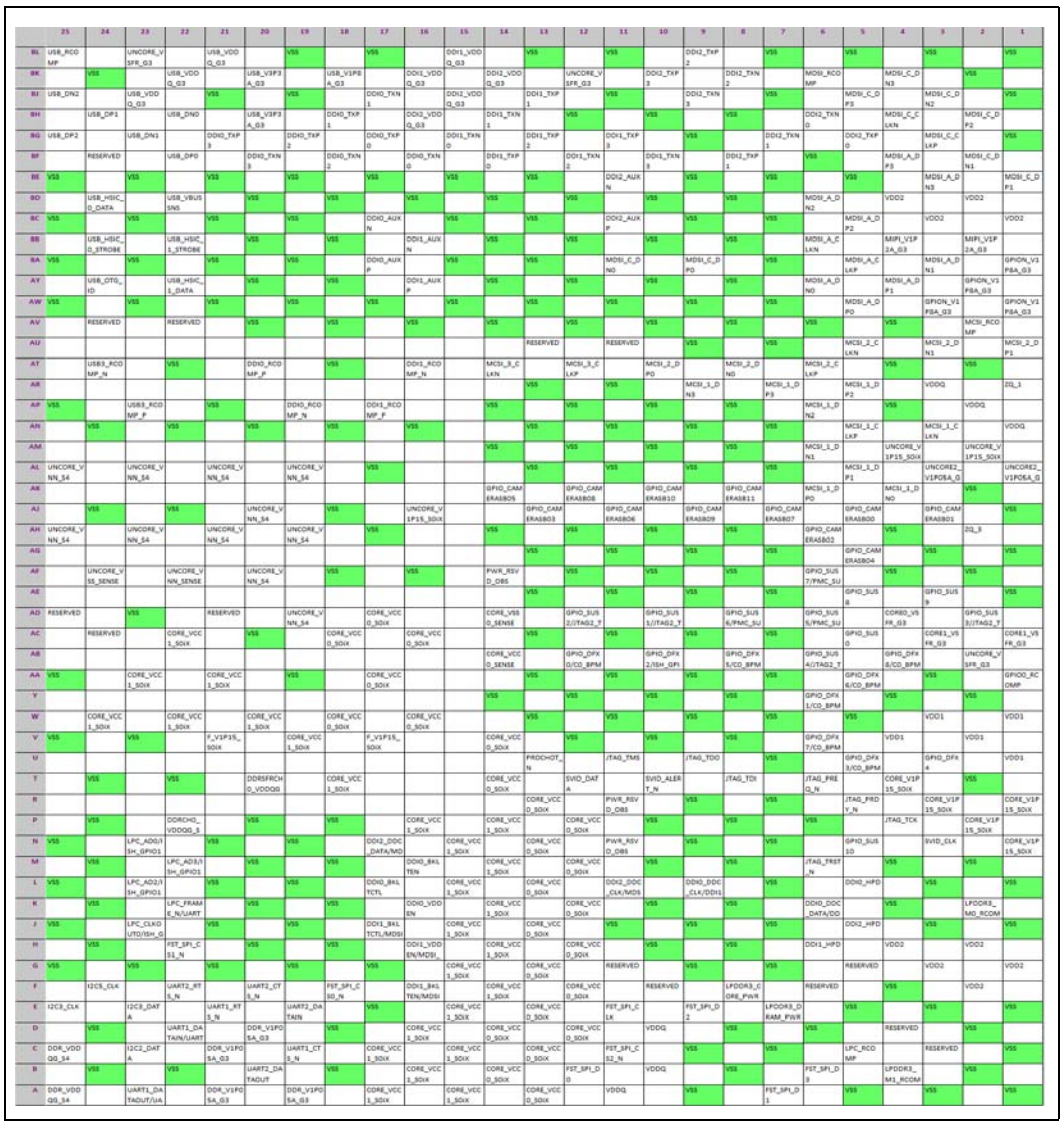


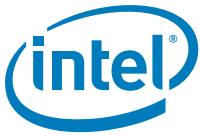
Figure 120.Ballout - Co-POP Part A



	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26
BL	VSS		VSS		VSS		PCIE_RXP0		ICLK_VSFR_03		2Q_4		PCIeCLK_V1P05A_03		ICLK_VSFR_03		USB_SSIC_RCOMP_P		PWR_RSVD_V1P05		USB3_V1P05A_03		VSS		USB3SIC_V1P2A_03	
BK		VSS		ICLK_OSCIN		PCIE_RXN0		UNCORE1_V1P05A_G		VSS		2Q_2		USB3_TXP0		USB_SSIC_RCOMP_N		PCIE_V1P05_G3	PWR_RSVD_V1P05		USB3SIC_V1P05A_G3		VSS		USB_HSIC_RCOMP	
BJ	VSS		ICLK_OSCOUT	RESERVED		PCIE_RXP1		ICLK_VSFR_03		VSS			PCIeCLK_V1P05A_03	USB3_TXN0		VSS		PWR_RSVD_V1P05		USB3_V1P05A_03		USB_SSIC1_RX_P		USBHSIC_V1P1A_03	
BH		ICLK_ICOMP	PCIE_REFCLKO_N		PCIE_RXN1		UNCORE1_V1P05A_G			VSS		VSS		RESERVED		VSS		PCIE_V1P05_G3		USB_SSIC0_RX_P		USB_SSIC1_RX_N		VSS		VSS
BG	VSS		PCIE_REFCLKO_P		VSS		VSS		PCIE_TXN0		PCIE_TXN1		RESERVED		USB3_TXN1		USB3_TXN2		USB3_TXP3		USB_SSIC0_RX_N		USB_SSIC0_TX_P		USB_SSIC1_TX_P	
BF		VDD2		PCIE_REFCLK1_N		PCIE_REFCLK1_P		PCIE_TXP0		PCIE_TXP1		RESERVED		RESERVED		USB3_TXP1		USB3_TXP2		USB3_TXN3		USB_SSIC0_TX_N		USB_SSIC1_TX_N		VSS
BE	VDD2		VDD2		RESERVED		VSS		VSS		RESERVED		VSS		VSS		USB3_RXN1		VSS		VSS		VSS		VSS	
BD		ICLK_RCOMP		VDD2		RESERVED		VSS		RESERVED		VSS		VSS		VSS		VSS		VSS			USB3_RXP3		VSS		VSS
BC	VDD1		PMC_RSMRST_N		RTC_RST_N		VSS		VSS		RESERVED		VSS		VSS		USB3_RXP1		VSS		VSS			USB_DP3		VSS	
BB		VDD1		VDD1		RTC_X2		VSS		VSS		VSS		VSS		USB3_RXP0		VSS		VSS		USB3_RXN3		VSS		VSS
BA	VDD1		VDD1		RESERVED		PMC_PLT_CLK0/ISH_		PMC_PLT_CLK1/ISH_		RTC_EXTPAD		VSS		VSS		USB3_RXN2		VSS		VSS		USB_DN3		VSS	
AY		VSS		VSS		RTC_X1		VSS		VSS		VSS		VSS		USB3_RXN0		VSS		VSS		VSS		VSS		VSS
AW	PMC_CORE_PWROK		RTC_TEST_N		VSS		VSS		VSS		VSS		VSS		VSS		USB3_RXP2		VSS		VSS			VSS		VSS	
AV		VDDQ		ISH_GPIO3/I2S0_DAT		ISH_GPIO1/I2S0_FS		VSS		VSS		VSS		VSS		F_V3P3A_G3		VSS		VSS		VSS		RESERVED		RESERVED
AU	VDDQ		VDDQ		ISH_GPIO4/I2S4_CLK		VSS		VSS		VSS		VSS														
AT		VDDQ		VDDQ		ISH_GPIO5/I2S4_FS		PMC_PLT_CLK5/ISH_		PMC_PLT_CLK2/ISH_		PMC_PLT_CLK4/ISH_		VSS			F_V1P8A_G3		F_V1P05A_G3		S_RCOMP_N		VSS		PCIE_RCOMP_N	VSS
AR	RTC_V3P3RTC_G5		ISH_GPIO0/I2S3_CLK		ISH_I2C1_CLK/ISH_3		VSS		RESERVED		PMC_PLT_CLK3/ISH_		VSS													
AP		RTC_V3P3A_G5		VSS		ISH_GPIO2/I2S3_DAT		VSS		VSS		VSS		VSS		VSS		F_V1P05A_G3		F_V1P05A_G3		S_RCOMP_P		VSS		PCIE_RCOMP_P	
AN	SDIO_V3P3A_V1P8A		VSS		ISH_GPIO6/I2S4_DAT		VSS		VSS		VSS		VSS				VSS		VSS		VSS		VSS		VSS	VSS
AM		LPC_V3P3A_V1P8A		VSS		ISH_I2C1_DATA/ISH_		VSS		VSS		VSS		VSS												
AL	ISH_GPIO8/I2S5_CLK		ISH_GPIO9/ISH_SPI_		ISH_GPIO7/I2S4_DAT		VSS		VSS		VSS		VSS			VSS		VSS		VSS		VSS		VSS	VSS	
AK		VSS		PMC_WAKE_N		PMC_SLP_S4_N		SPI1_MOSI		SPI1_MISO		SPI1_CLK		DDI_VSS_SENSE												
AJ	GPIO5E_V1P8A_G3		GPIO5E_V1P8A_G3		PMC_SUS_STAT_N		VSS		VSS		VSS		VSS				UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4	UNCORE_VNN_S4
AH		GPIO5E_V1P8A_G3	GPIO5E_V1P8A_G3		PMC_PWRBTN_N		VSS		SPI1_CS1_N		SPI1_CS0_N		VGG_SENSE		UNCORE_VNN_S4		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS	VSS	
AG	VSS		PMC_SUS_S3_N		PMC_PLTRST_N		VSS		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4													
AF		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4			UNCORE_VNN_S4		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		VSS	VSS
AE	UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4		UNCORE_VNN_S4													
AD		VSS		VSS		PWM0		VSS		VSS		VSS		VSS		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS	VSS	
AC	PMC_SLP_SDIX_N		PMC_SUSCLK0		PMC_ACPRESENT		PWM1/I6H_GPIO10		VSS		VSS		VSS				VSS		DDI_VGG_SDIX		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX	VSS
AB		VSS		PMC_BATLOW_N		SD3_D3		VSS		VSS		VSS		VSS												
AA	UNCORE_V1P8A_G3		VSS		MMC1_CMD		VSS		VSS		VSS		VSS			VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		DDI_VGG_SDIX	VSS	
Y		VSS		VSS		SD3_CLK		VSS		VSS		VSS		PWR_RSVD_D86												
W	VSS		MMC1_D1		MMC1_D3		VSS		VSS		VSS		VSS				VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX	CORE_VCCS_SENSE
V		VSS		VSS		SD3_D2		VSS		VSS		VSS		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX	CORE_VSS_SENSE
U	VSS		VSS		SD3_D1		VSS		VSS		VSS		VSS													
T		VSS		VSS		SD3_CMD		VSS		VSS		VSS		DDI_VGG_SDIX			DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		VSS		VSS	PWR_RSVD_C85
R	MMC1_RCOMP		SD2_CMD		SD2_D2		VSS		VSS		VSS		DDI_VGG_SDIX													
P		DDI_V1P15_SDIX		DDI_V1P15_SDIX		MMC1_CLK		VSS		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		VSS		DDRCH1_VDDQS_S		LPE_AD1/ISH_GPIO1		VSS
N	DDI_V1P15_SDIX		DDI_V1P15_SDIX		MMC1_D2		VSS		VSS		VSS		DDI_VGG_SDIX	DDI_VGG_SDIX			DDRSFRCH1_VDDQS		VSS		VSS		VSS		VSS	
M		SD2_CLK		DDI_V1P15_SDIX		MMC1_D0		VSS		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		VSS		VSS		USB_OC0_N		LPC_CLKOUT1/ISH_0		VSS
L	VSS		SD2_D1		SD2_D0		MMC1_D4		VSS		LPE_I2S2_FRM		DDI_VGG_SDIX		VSS		LPE_I2S0_FRM		VSS		VSS		VSS		VSS	
K		VSS		VSS		VSS		VSS		VSS		DDI_VGG_SDIX		DDI_VGG_SDIX		I2C4_DATA/CCD2_D		VSS		VSS		USB_OC1_N		LPC_CLKRUN_N/UART		VSS
J	SD3_RCOMP		VSS		SD3_D3_CD_N		VSS		VSS		PCIE_CLKREQ1_N		DDI_VGG_SDIX		VSS		I2C4_CLK/CCD2_SDC		VSS		VSS		VSS		VSS	
H		VSS		VSS		SD3_D0		VSS		LPE_I2S0_CLK		DDI_VGG_SDIX		DDI_VGG_SDIX		I2C0_CLK		VSS		VSS		NFC_I2C_DATA		NFC_I2C_CLK		VSS
G	VSS		MMC1_D6		MMC1_D5		VSS		LPE_I2S1_FRM		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		VSS		VSS		VSS		VSS		VSS	
F		MMC1_D7		VSS		PMC_RSTBTN_N		LPE_I2S3_DATAIN		DDI_VGG_SDIX		DDI_VGG_SDIX		LPE_I2S2_DATAIN		MMC1_RESET_N		UART0_DATAIN		SPI3_MISO		I2C5_DATA		I2C6_CLK/NMI_N		I2C6_DATA/SD3_WP
E	VSS		MMC1_RCLK		VSS		LPE_I2S1_CLK		LPE_I2S0_DATAOUT		DDI_VGG_SDIX		GPIO_SW93		LPE_I2S2_CLK		SPI3_MOSI		ISH_GPIO12/ISH_UAR		SPI3_CS0_N		I2C1_DATA		I2C2_CLK	
D		GPIO_ALERT/ISH_GP		SD3_1P8_EN		LPC_SERIRQ/SPI2_C5		VDD2		DDI_VGG_SDIX		DDI_VGG_SDIX		SD3_WP		DDR_V1P05A_G3		VDDQ		VSS		I2C0_DATA		DDR_VDDQ0_S4		DDR_VDDQG_S4
C	VSS		SD3_CD_N		SD3_PWREN_N		VDD2		LPE_I2S0_DATAIN		DDI_VGG_SDIX		PCIE_CLKREQ0_N		DDR_V1P05A_G3		GPIO_SW78		VDDQ		UART0_DATAOUT/SPI		DDR_VDDQ0_S4		I2C1_CLK	
B		VSS		PMC_SUSWRDNACK		VDD2		LPE_I2S3_DATAOUT		DDI_VGG_SDIX		DDI_VGG_SDIX		DDR_V1P05A_G3		DDR_V1P05A_G3		VDDQ		VSS		DDR_VDDQ0_S4		VSS		DDR_VDDQG_S4
A	VSS		VSS		VSS		VDD2		DDI_VGG_SDIX		DDI_VGG_SDIX		VSS		DDR_V1P05A_G3		VDDQ		VDDQ		LPE_I2S2_DATAOUT		DDR_VDDQ0_S4		DDR_VDDQG_S4	

Figure 121.Ballout - Co-POP Part B





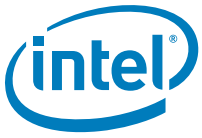
21.2 SoC VMS-T3 Pin List Location

Ball #	DDR3L-RS Customer Pin List
A10	DDI0_DDC_CLK
A12	CORE_VCC_S0iX
A13	GPIO_SUS9
A14	GPIO_SUS0
A16	CORE_V1P15_S0iX
A17	DDR_VDDQG_S4
A18	DDR3_M0_DQ[6]
A2	RESERVED
A20	DDR3_M0_DM[0]
A21	DDR3_M0_DQ[2]
A22	DDR3_M0_DQ[4]
A23	DDR3_M0_DQ[17]
A24	VSS
A25	PWR_RSVD_OBS
A3	MCSI_2_DP[1]
A4	MCSI_2_DP[0]
A5	MDSI_RCOMP
A6	MCSI_1_DP[1]
A8	MCSI_RCOMP
A9	DDI0_VDDEN
AA1	ICLK_RCOMP
AA10	SD3_CMD
AA11	VSS
AA12	FST_SPI_CS[0]_N
AA13	LPC_SERIRQ/SPI2_CS[0]_N
AA14	LPE_I2S1_FRM
AA15	LPE_I2S0_DATAOUT
AA16	LPE_I2S0_CLK
AA17	I2C2_CLK
AA18	VSS
AA19	DDR3_DRAM_PWROK
AA2	VSS
AA20	DDR3_CORE_PWROK

Ball #	DDR3L-RS Customer Pin List
AA21	VSS
AA22	DDR3_M0_DQ[52]
AA23	DDR3_M0_DQS[6]_N
AA24	VSS
AA25	DDR3_M0_DQ[36]
AA3	ISH_I2C1_CLK/ISH_SPI_CLK/ I2S5_DATAIN
AA4	ISH_GPIO[9]/ISH_SPI_MISO/ I2S5_FS
AA5	ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT
AA6	MMC1_CMD
AA7	MMC1_D[7]
AA8	MMC1_D[0]
AA9	SD3_PWREN_N
AB1	ICLK_ICOMP
AB10	SD3_D[2]
AB11	FST_SPI_D[0]
AB12	FST_SPI_CLK
AB13	UART2_DATAOUT
AB14	LPE_I2S1_CLK
AB15	LPE_I2S0_DATAIN
AB16	I2C5_CLK
AB17	I2C4_CLK/DDI2_DDC_CLK
AB18	I2C4_DATA/DDI2_DDC_DATA
AB19	LPE_I2S2_CLK
AB2	ISH_GPIO[0]/I2S3_CLK
AB20	PCIE_CLKREQ[0]_N/ GPIO_SW90
AB21	RESERVED
AB22	DDR3_M0_DQ[48]
AB23	DDR3_M0_DQS[6]_P
AB24	DDR3_M0_DQS[4]_N
AB25	DDR3_M0_DQ[33]
AB3	ISH_GPIO[2]/I2S3_DATAOUT

Ball #	DDR3L-RS Customer Pin List
AB4	PMC_PLT_CLK[3]/ ISH_GPIO[13]/ ISH_UART_RTS_N
AB5	SD2_CMD
AB6	MMC1_D[2]
AB7	MMC1_D[6]
AB8	SD3_1P8_EN
AB9	SD3_D[1]
AC1	ISH_GPIO[7]/I2S4_DATAIN
AC10	SD3_D[0]
AC11	FST_SPI_D[1]
AC12	UART2_RTS_N
AC13	UART1_CTS_N
AC14	UART1_DATAOUT/ UART0_DATAOUT
AC15	NFC_I2C_DATA
AC16	NFC_I2C_CLK
AC17	I2C2_DATA
AC18	GPIO_SW78
AC19	LPE_I2S2_FRM
AC2	ISH_GPIO[4]/I2S4_CLK
AC20	GPIO_SW93
AC21	RESERVED
AC22	DDR3_M0_DQ[49]
AC23	DDR3_M0_DQ[37]
AC24	DDR3_M0_DM[4]
AC25	DDR3_M0_DQS[4]_P
AC3	PMC_PLT_CLK[4]/ ISH_GPIO[14]/ ISH_I2C0_DATA
AC4	PMC_PLT_CLK[2]/ ISH_GPIO[12]/ ISH_UART_CTS_N
AC5	VSS
AC6	MMC1_D[3]
AC7	MMC1_D[4]
AC8	MMC1_D[5]
AC9	SD3_CLK
AD1	RESERVED

Ball #	DDR3L-RS Customer Pin List
AD10	SD3_D[3]
AD12	UART2_CTS_N
AD13	VSS
AD14	UART1_DATAIN/ UART0_DATAIN
AD16	I2C6_DATA/SD3_WP
AD17	VSS
AD18	MMC1_RESET_N
AD2	PWM[1]/ISH_GPIO[10]/ ISH_UART_DATAOUT
AD20	SD3_WP
AD21	DDR3_M0_DQ[51]
AD22	VSS
AD23	DDR3_M0_DQ[38]
AD24	DDR3_M0_DQ[32]
AD25	VSS
AD3	PWM[0]
AD4	SD2_D[0]
AD5	SD2_CLK
AD6	SD2_D[3]_CD_N
AD8	MMC1_CLK
AD9	VSS
AE1	RESERVED
AE10	MMC1_RCOMP
AE12	UART2_DATAIN
AE13	UART1_RTS_N
AE14	SD3_RCOMP
AE16	I2C6_CLK/NMI_N
AE17	I2C5_DATA
AE18	UART0_DATAIN
AE2	RESERVED
AE20	DDR3_M0_RCOMPPD
AE21	DDR3_M0_OCAVREF
AE22	DDR3_M0_ODQVREF
AE23	DDR3_M0_DQ[35]
AE24	PWR_RSVD_OBS
AE25	PWR_RSVD_OBS

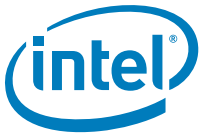


Ball #	DDR3L-RS Customer Pin List
AE3	PMC_PLT_CLK[5]/ ISH_GPIO[15]/ISH_I2C0_CLK
AE4	SD2_D[1]
AE5	SD2_D[2]
AE6	DDI_VGG_S0iX
AE8	MMC1_RCLK
AE9	SD3_CD_N
B1	RESERVED
B10	GPIO_CAMERASB10
B12	GPIO_SUS7/PMC_SUSCLK[3]
B13	VSS
B14	GPIO_SUS3/JTAG2_TDI
B16	GPIO0_RCOMP
B17	VSS
B18	DDR3_M0_DQ[1]
B2	MDSI_A_DN[0]
B20	DDR3_M0_DQ[5]
B21	VSS
B22	DDR3_M0_DQ[7]
B23	DDR3_M0_DQ[22]
B24	DDR3_M0_DQ[20]
B25	PWR_RSVD_OBS
B3	MCSI_2_DN[1]
B4	MCSI_2_DN[0]
B5	VSS
B6	MCSI_1_DN[1]
B8	DDI0_BKLTCTL
B9	VSS
C1	MDSI_A_DN[1]
C10	DDI0_DDC_DATA
C11	GPIO_CAMERASB11
C12	JTAG_TDI
C13	GPIO_SUS8
C14	GPIO_SUS6/PMC_SUSCLK[2]
C15	GPIO_SUS1/JTAG2_TCK
C16	GPIO_DFX8/C0_BPM3_TX/ C1_BPM3_TX

Ball #	DDR3L-RS Customer Pin List
C17	GPIO_DFX1/C0_BPM1_TX/ C1_BPM1_TX
C18	DDR3_M0_DQ[3]
C19	DDR3_M0_DQS[0]_N
C2	MDSI_A_DP[1]
C20	DDR3_M0_DQ[8]
C21	DDR3_M0_DQS[1]_N
C22	DDR3_M0_DQS[1]_P
C23	DDR3_M0_DQ[18]
C24	DDR3_M0_DQS[2]_P
C25	DDR3_M0_DQS[2]_N
C3	MDSI_A_DP[0]
C4	MDSI_A_CLKP
C5	MCSI_2_CLKN
C6	MCSI_1_DN[3]
C7	MCSI_1_DP[2]
C8	DDI0_BKLTEN
C9	DDI2_DDC_DATA/ MDSI_C_TE/UART0_DATAIN
D1	DDI1_RCOMP_N
D10	GPIO_CAMERASB09
D11	GPIO_CAMERASB08
D12	VSS
D13	JTAG_PRDY_N
D14	GPIO_SUS4/JTAG2_TDO
D15	GPIO_SUS2/JTAG2_TMS
D16	GPIO_DFX6/C0_BPM1_TX/ C1_BPM1_TX
D17	GPIO_DFX2/ISH_GPIO[13]/ C0_BPM2_TX
D18	DDR3_M0_DQ[0]
D19	DDR3_M0_DQS[0]_P
D2	DDI1_RCOMP_P
D20	DDR3_M0_DQ[15]
D21	DDR3_M0_DM[1]
D22	DDR3_M0_DQ[12]
D23	DDR3_M0_DQ[9]
D24	DDR3_M0_DQ[21]

Ball #	DDR3L-RS Customer Pin List
D25	DDR3_M0_DM[2]
D3	MDSI_A_DP[2]
D4	MDSI_A_CLKN
D5	MCSI_2_CLKP
D6	MCSI_1_DP[3]
D7	MCSI_1_DN[2]
D8	MDSI_A_TE/MDSI_C_TE
D9	DDI0_HPD
E1	DDI0_AUXP
E10	DDI2_DDC_CLK/MDSI_A_TE/ UART0_DATAOUT
E11	VSS
E12	SVID_ALERT_N
E13	JTAG_TRST_N
E14	JTAG_TCK
E15	VSS
E16	GPIO_DFX5/C0_BPM0_TX/ C1_BPM0_TX
E17	GPIO_DFX0/C0_BPM0_TX
E18	VSS
E19	DDR3_M0_DQ[10]
E2	VSS
E20	DDR3_M0_DQ[14]
E21	VSS
E22	DDR3_M0_DQ[13]
E23	DDR3_M0_DQ[11]
E24	VSS
E25	DDR3_M0_DQ[16]
E3	MDSI_A_DN[2]
E4	MDSI_A_DP[3]
E5	MDSI_A_DN[3]
E6	VSS
E7	VSS
E8	MCSI_1_DP[0]
E9	DDI2_HPD
F1	DDI0_AUXN
F10	RESERVED

Ball #	DDR3L-RS Customer Pin List
F11	SVID_CLK
F12	SVID_DATA
F13	JTAG_TDO
F14	JTAG_TMS
F15	VSS
F16	GPIO_DFX7/C0_BPM2_TX
F17	GPIO_DFX3/C0_BPM3_TX
F18	GPIO_DFX4
F19	DDR3_M0_DM[3]
F2	DDI0_TXN[3]
F20	DDR3_M0_DQ[24]
F21	DDR3_M0_DQ[25]
F22	DDR3_M0_CKE[3]
F23	DDR3_M0_CKE[2]
F24	DDR3_M0_DQ[23]
F25	DDR3_M0_DQ[19]
F3	DDI0_TXP[3]
F4	RESERVED
F5	DDI0_TXP[2]
F6	DDI0_TXN[2]
F7	MCSI_1_CLKP
F8	MCSI_1_DN[0]
F9	PROCHOT_N
G10	VSS
G11	VSS
G12	RESERVED
G13	UNCORE_V1P15_S0iX
G14	CORE_VCC_S0iX
G15	CORE_VCC_S0iX
G16	VSS
G17	CORE_V1P15_S0iX
G18	CORE_V1P15_S0iX
G19	DDR_V1P05A_G3
G20	DDR3_M0_DQ[26]
G21	DDR3_M0_DQ[28]
G22	DDR3_M0_CKE[0]
G23	DDR3_M0_CKE[1]

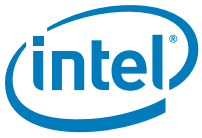


Ball #	DDR3L-RS Customer Pin List
G3	DDI0_TXP[0]
G4	DDI0_TXN[0]
G5	VSS
G6	VSS
G7	MCSI_1_CLKN
G8	UNCORE_VSFR_G3
G9	MIPI_V1P2A_G3
H1	DDI2_TXN[2]
H10	VSS
H11	USB_V1P8A_G3
H12	USB_V1P8A_G3
H13	UNCORE_V1P15_S0iX
H14	CORE0_VSFR_G3
H15	CORE_VCC_S0iX
H16	VSS
H17	VSS
H18	DDR_V1P05A_G3
H19	DDR3_M0_DQS[3]_N
H2	DDI2_TXP[2]
H20	DDR3_M0_DQS[3]_P
H21	VSS
H22	DDR3_M0_DQ[31]
H23	DDR3_M0_DQ[27]
H24	DDR3_M0_MA[09]
H25	DDR3_M0_MA[14]
H3	DDI0_RCOMP_P
H4	DDI0_RCOMP_N
H5	DDI2_AUXP
H6	DDI0_TXP[1]
H7	USB_VDDQ_G3
H8	DDI_USB_VDDQ_G3
H9	MIPI_V1P2A_G3
J1	USB_RCOMP
J10	UNCORE_VNN_S4
J11	VSS
J12	VSS
J13	VSS

Ball #	DDR3L-RS Customer Pin List
J14	CORE_VCC_S0iX
J15	CORE_VCC_S0iX
J16	VSS
J17	VSS
J18	VSS
J19	VSS
J2	VSS
J20	DDR3_M0_DQ[30]
J21	DDR3_M0_DQ[29]
J22	DDR3_M0_CK[0]_P
J23	DDR3_M0_CK[0]_N
J24	VSS
J25	DDR3_M0_MA[8]
J3	DDI2_TXP[0]
J4	DDI2_TXN[0]
J5	DDI2_AUXN
J6	DDI0_TXN[1]
J7	DDI_USB_VDDQ_G3
J8	USBHSIC_V1P2A_G3
J9	UNCORE_VNN_S4
K1	USB_OTG_ID
K10	UNCORE_VNN_S4
K11	UNCORE_VNN_S4
K12	UNCORE_VNN_S4
K13	VSS
K14	VSS
K15	CORE_VCC_S0iX
K16	F_V1P15_S0iX
K17	CORE_V1P15_S0iX
K18	DDR_VDDQG_S4
K19	VSS
K2	USB_VBUSSNS
K20	DDR3_M0_CK[1]_P
K21	DDR3_M0_CK[1]_N
K22	DDR3_M0_BS[2]
K23	DDR3_M0_MA[15]
K24	DDR3_M0_MA[6]

Ball #	DDR3L-RS Customer Pin List
K25	DDR3_M0_MA[11]
K3	DDI2_TXN[1]
K4	DDI2_TXP[1]
K5	DDI2_TXP[3]
K6	DDI2_TXN[3]
K7	DDI_USB_VDDQ_G3
K8	USBSSIC_V1P2A_G3
K9	UNCORE_VNN_S4
L10	UNCORE_VNN_S4
L11	UNCORE_VNN_S4
L12	UNCORE_VNN_S4
L13	UNCORE2_V1P05A_G3
L14	CORE_VCC_S0iX
L15	CORE_VCC_S0iX
L16	F_V1P15_S0iX
L17	CORE_V1P15_S0iX
L18	DDR_VDDQG_S4
L19	DDR_VDDQG_S4
L20	DDR_VDDQG_S4
L21	VSS
L22	DDR3_M0_MA[5]
L23	DDR3_M0_MA[1]
L3	USB_DN[1]
L4	USB_DP[1]
L5	VSS
L6	UNCORE_VSFR_G3
L7	VSS
L8	USB_V3P3A_G3
L9	UNCORE_VNN_S4
M1	USB_HSIC_0_STROBE
M10	UNCORE_VNN_S4
M11	UNCORE_VNN_S4
M12	UNCORE_VSFR_G3
M13	VSS
M14	CORE_VCC_S0iX
M15	CORE_VCC_S0iX
M16	VSS

Ball #	DDR3L-RS Customer Pin List
M17	VSS
M18	DDR_VDDQG_S4
M19	VSS
M2	USB_HSIC_0_DATA
M20	DDR3_M0_MA[3]
M21	DDR3_M0_BS[0]
M22	DDR3_M0_MA[4]
M23	DDR3_M0_MA[13]
M24	DDR3_M0_MA[0]
M25	DDR3_M0_MA[12]
M3	USB_DN[3]
M4	USB_DP[3]
M5	USB_DP[2]
M6	USB_DN[2]
M7	USBSSIC_V1P05A_G3
M8	VSS
M9	UNCORE_VNN_S4
N1	USB_HSIC_RCOMP
N10	UNCORE_VNN_S4
N11	DDI_VGG_S0iX
N12	DDI_VGG_S0iX
N13	VSS
N14	CORE_VCC_S0iX
N15	CORE_VCC_S0iX
N16	VSS
N17	VSS
N18	VSS
N19	DDR3_M0_BS[1]
N2	VSS
N20	DDR3_M0_MA[10]
N21	VSS
N22	DDR3_M0_RAS_N
N23	DDR3_M0_MA[2]
N24	VSS
N25	DDR3_M0_MA[7]
N3	USB_HSIC_1_STROBE
N4	USB_HSIC_1_DATA

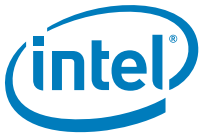


Ball #	DDR3L-RS Customer Pin List
N5	USB3_RXP[0]
N6	USB_DP[0]
N7	MPHY_1P05A_G3
N8	MPHY_1P05A_G3
N9	VSS
P1	USB3_TXP[0]
P10	DDI_VGG_S0iX
P11	DDI_VGG_S0iX
P12	VSS
P13	RESERVED
P14	VSS
P15	VSS
P16	VSS
P17	VSS
P18	VSS
P19	DDRSFR_VDDQG_S4
P2	USB3_TXN[0]
P20	DDR3_M0_CAS_N
P21	DDR3_M0_WE_N
P22	DDR3_M0_CS[0]_N
P23	DDR3_M0_CS[1]_N
P24	DDR3_M0_ODT[0]
P25	DDR3_M0_ODT[1]
P3	USB3_RCOMP_P
P4	USB3_RCOMP_N
P5	USB3_RXN[0]
P6	USB_DN[0]
P7	PCIeCLK_V1P05A_G3
P8	VSS
P9	UNCORE_VNN_S4
R10	VSS
R11	DDI_VGG_S0iX
R12	VSS
R13	DDI_VGG_S0iX
R14	DDI_VGG_S0iX
R15	VSS
R16	DDI_V1P15_S0iX

Ball #	DDR3L-RS Customer Pin List
R17	DDI_V1P15_S0iX
R18	DDR_V1P05A_G3
R19	VSS
R20	VSS
R21	VSS
R22	DDR3_M0_DQ[45]
R23	DDR3_M0_DQ[44]
R3	PCIE_TXN[0]
R4	PCIE_TXP[0]
R5	VSS
R6	VSS
R7	F_V1P05A_G3
R8	ICLK_VSFR_G3
R9	VSS
T1	PCIE_RXP[0]
T10	UNCORE1_V1P05A_G3
T11	DDI_VGG_S0iX
T12	DDI_VGG_S0iX
T13	DDI_VGG_S0iX
T14	DDI_VGG_S0iX
T15	DDI_VGG_S0iX
T16	DDI_VGG_S0iX
T17	DDI_VGG_S0iX
T18	VSS
T19	DDR_VDDQG_S4
T2	PCIE_RXN[0]
T20	DDR3_M0_DQ[57]
T21	DDR3_M0_DQ[63]
T22	DDR3_M0_DQ[56]
T23	DDR3_M0_DM[7]
T24	DDR3_M0_DQ[41]
T25	DDR3_M0_DQ[47]
T3	PCIE_RCOMP_P
T4	PCIE_RCOMP_N
T5	PMC_RSMRST_N
T6	PMC_CORE_PWROK
T7	F_V1P05A_G3

Ball #	DDR3L-RS Customer Pin List
T8	VSS
T9	F_V1P05A_G3
U1	RTC_EXTPAD
U10	DDI_VGG_S0iX
U11	DDI_VGG_S0iX
U12	DDI_VGG_S0iX
U13	DDI_VGG_S0iX
U14	DDI_VGG_S0iX
U15	DDI_VGG_S0iX
U16	DDI_VGG_S0iX
U17	LPE_I2S2_DATAIN
U18	DDR_V1P05A_G3
U19	DDR3_M0_DRAMRST_N
U2	VSS
U20	DDR3_M0_DQ[62]
U21	DDR3_M0_DQ[60]
U22	DDR3_M0_DQS[7]_P
U23	DDR3_M0_DQS[7]_N
U24	VSS
U25	DDR3_M0_DQS[5]_N
U3	RTC_X1
U4	RTC_X2
U5	RTC_RST_N
U6	RTC_TEST_N
U7	F_V3P3A_G3
U8	RTC_V3P3A_G5
U9	F_V1P8A_G3
V1	UNCORE_VNN_S4
V10	UNCORE_V1P8A_G3
V11	PMC_RSTBTN_N
V12	PMC_SUSPWRDNACK
V13	LPC_FRAME_N/ UART0_DATAIN/SPI2_MISO
V14	DDI_VGG_S0iX
V15	UNCORE_V1P8A_G3
V16	UNCORE_V1P8A_G3
V17	DDI_VGG_S0iX

Ball #	DDR3L-RS Customer Pin List
V18	LPE_I2S2_DATAOUT
V19	DDR_VDDQG_S4
V2	PCIE_REFCLK[0]_N
V20	DDR3_M0_DQ[59]
V21	VSS
V22	DDR3_M0_DQ[46]
V23	DDR3_M0_DQ[40]
V24	DDR3_M0_DM[5]
V25	DDR3_M0_DQS[5]_P
V3	PCIE_REFCLK[0]_P
V4	RESERVED
V5	PMC_SUS_STAT_N
V6	VSS
V7	VSS
V8	RTC_V3P3RTC_G5
V9	VSS
W10	RESERVED
W11	VSS
W12	VSS
W13	SDIO_V3P3A_V1P8A_G3
W14	DDI_VGG_S0iX
W15	I2C1_DATA
W16	I2C1_CLK
W17	VSS
W18	VSS
W19	VSS
W20	DDR3_M0_DQ[58]
W21	DDR3_M0_DQ[61]
W22	DDR3_M0_DQ[42]
W23	DDR3_M0_DQ[43]
W3	PMC_PWRBTN_N
W4	PMC_SLP_S0IX_N
W5	VSS
W6	ICLK_OSCOUT
W7	ICLK_OSCIN
W8	VSS
W9	UNCORE_V1P8A_G3



Ball #	DDR3L-RS Customer Pin List
Y1	ISH_GPIO[3]/I2S3_DATAIN
Y10	RESERVED
Y11	VSS
Y12	LPC_CLKRUN_N/ UART0_DATAOUT/SPI2_CLK
Y13	LPE_I2S1_DATAIN
Y14	LPE_I2S1_DATAOUT
Y15	DDI_VGG_S0iX
Y16	LPE_I2S0_FRM
Y17	I2C0_CLK
Y18	I2C0_DATA
Y19	DDR3_M0_DQ[50]
Y2	ISH_GPIO[1]/I2S3_FS
Y20	DDR3_M0_DQ[54]
Y21	DDR3_M0_DQ[55]
Y22	DDR3_M0_DM[6]
Y23	DDR3_M0_DQ[53]
Y24	DDR3_M0_DQ[34]
Y25	DDR3_M0_DQ[39]
Y3	PMC_WAKE_N
Y4	PMC_SUSCLK[0]
Y5	PMC_PLTRST_N
Y6	VSSA
Y7	VSS
Y8	MMC1_D[1]
Y9	USB_OC[0]_N

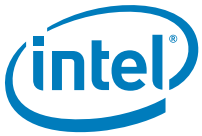
21.3 SoC MSP-T4 Pin List Location

Ball #	Customer Name - LPDDR3
A11	MIPI_V1P2A_G3
A13	VSS
A15	GPION_V1P8A_G3
A17	VSS

Ball #	Customer Name - LPDDR3
A19	CORE_VCC0_S0iX
A21	CORE_VCC0_S0iX
A23	CORE_VCC1_S0iX
A25	CORE_VCC1_S0iX
A27	VSS
A3	VSS

Ball #	Customer Name - LPDDR3
A30	CORE_V1P15_S0iX
A33	CORE_V1P15_S0iX
A35	CORE_V1P15_S0iX
A37	CORE_V1P15_S0iX
A39	VSS
A41	DDR_V1P05A_G3
A43	DDR_V1P05A_G3
A45	VSS
A47	VSS
A49	VSS
A5	VSS
A51	VSS
A53	VSS
A55	VSS
A57	VSS
A7	VSS
A9	MIPI_V1P2A_G3
AA1	USBHSIC_V1P2A_G3
AA11	VSS
AA15	VSS
AA17	VSS
AA19	UNCORE_VNN_S4
AA21	UNCORE_VNN_S4
AA23	UNCORE_VNN_S4
AA25	UNCORE_V1P15_S0iX
AA27	UNCORE_VNN_S4
AA29	UNCORE_VNN_S4
AA3	VSS
AA32	VSS
AA34	UNCORE_VNN_S4
AA36	UNCORE_VNN_S4
AA38	VSS
AA40	VSS
AA42	DDR_V1P05A_G3
AA44	VSS
AA46	VSS
AA49	VSS

Ball #	Customer Name - LPDDR3
AA5	USB_DN[3]
AA51	LPDDR3_M0_DQ[18]_B
AA53	LPDDR3_M0_DQ[21]_B
AA55	LPDDR3_M0_DQ[07]_B
AA57	LPDDR3_M0_DQ[08]_B
AA59	DDR_VDDQG_S4
AA7	USB_HSIC_0_STROBE
AA9	USB_HSIC_0_DATA
AB10	USB_HSIC_1_STROBE
AB12	USB_RCOMP
AB2	USBHSIC_V1P2A_G3
AB4	USB_SSIC1_RX_P
AB48	LPDDR3_M0_DM[03]_B
AB50	LPDDR3_M0_DM[02]_B
AB52	LPDDR3_M0_DQ[23]_B
AB54	LPDDR3_M0_DQ[10]_B
AB56	LPDDR3_M0_DQ[09]_B
AB58	VSS
AB6	VSS
AB8	USB_HSIC_1_DATA
AC1	VSS
AC11	VSS
AC14	VSS
AC16	VSS
AC18	USB_V1P8A_G3
AC20	UNCORE_VNN_S4
AC22	VSS
AC24	UNCORE_VSS_SENSE
AC26	UNCORE_VNN_SENSE
AC28	RESERVED
AC3	USB_SSIC0_RX_P
AC31	UNCORE_VNN_S4
AC33	UNCORE_VNN_S4
AC35	VSS
AC37	VSS
AC39	VSS
AC41	DDRSFRCH0_VDDQG_S4

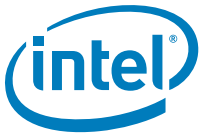


Ball #	Customer Name - LPDDR3
AC43	VSS
AC45	VSS
AC49	VSS
AC5	USB_SSIC1_RX_N
AC51	VSS
AC53	VSS
AC55	LPDDR3_M0_DQ[14]_B
AC57	LPDDR3_M0_DQ[13]_B
AC59	DDR_VDDQG_S4
AC7	VSS
AC9	VSS
AD10	USB_SSIC1_TX_N
AD12	USB_HSIC_RCOMP
AD15	VSS
AD17	VSS
AD19	UNCORE_VNN_S4
AD2	RESERVED
AD21	UNCORE_VNN_S4
AD23	VSS
AD25	UNCORE_VNN_S4
AD27	UNCORE_VNN_S4
AD29	UNCORE_VSFR_G3
AD32	CORE_VCC1_S0iX
AD34	CORE_VCC1_S0iX
AD36	CORE_VCC1_S0iX
AD38	CORE_VCC1_S0iX
AD4	USB_SSIC0_RX_N
AD40	F_V1P15_S0iX
AD42	DDRCH0_VDDQG_S4
AD44	LPDDR3_M0_RCOMPPD
AD46	VSS
AD48	LPDDR3_M0_DQ[25]_B
AD50	LPDDR3_M0_DQ[26]_B
AD52	VSS
AD54	VSS
AD56	VSS
AD58	LPDDR3_M0_DQS[1]_N_B

Ball #	Customer Name - LPDDR3
AD6	VSS
AD8	VSS
AE1	USBSSIC_V1P05A_G3
AE11	USB_SSIC1_TX_P
AE3	RESERVED
AE49	LPDDR3_M0_DQ[30]_B
AE5	USB3_RXP[3]
AE51	LPDDR3_M0_DQS[3]_N_B
AE53	LPDDR3_M0_DQ[27]_B
AE55	LPDDR3_M0_DQ[12]_B
AE57	LPDDR3_M0_DQS[1]_P_B
AE59	DDR_VDDQG_S4
AE7	VSS
AE9	USB_SSIC0_TX_P
AF10	USB_SSIC0_TX_N
AF12	VSS
AF14	UNCORE_VSFR_G3
AF16	VSS
AF18	USBSSIC_V1P2A_G3
AF2	VSS
AF20	UNCORE_VNN_S4
AF22	UNCORE_VNN_S4
AF24	UNCORE2_V1P05A_G3
AF26	UNCORE2_V1P05A_G3
AF28	RESERVED
AF31	VSS
AF33	CORE_VCC1_S0iX
AF35	CORE_VCC1_S0iX
AF37	CORE_VCC1_S0iX
AF39	CORE_VCC1_SENSE
AF4	USB3_RXN[3]
AF41	CORE_VSS1_SENSE
AF43	VSS
AF45	RESERVED
AF48	LPDDR3_M0_DQ[29]_B
AF50	LPDDR3_M0_DQS[3]_P_B
AF52	LPDDR3_M0_DQ[31]_B

Ball #	Customer Name - LPDDR3
AF54	VSS
AF56	LPDDR3_M0_DM[1]_B
AF58	VSS
AF6	VSS
AF8	USB_SSIC_RCOMP_P
AG1	USB3_V1P05A_G3
AG11	VSS
AG3	USB3_RXN[1]
AG49	VSS
AG5	VSS
AG51	VSS
AG53	VSS
AG55	LPDDR3_M0_DQ[11]_B
AG57	LPDDR3_M0_DQ[15]_B
AG59	VSS
AG7	VSS
AG9	USB_SSIC_RCOMP_N
AH10	VSS
AH12	VSS
AH15	VSS
AH17	VSS
AH19	UNCORE_VNN_S4
AH2	USB3_RXP[1]
AH21	UNCORE_VNN_S4
AH23	UNCORE_VNN_S4
AH25	UNCORE_VNN_S4
AH27	RESERVED
AH29	VSS
AH32	CORE_VCC1_S0iX
AH34	CORE_VCC1_S0iX
AH36	CORE_VCC1_S0iX
AH38	CORE_VCC1_S0iX
AH4	USB3_RCOMP_N
AH40	VSS
AH42	VSS
AH44	VSS
AH46	RESERVED

Ball #	Customer Name - LPDDR3
AH48	LPDDR3_M0_DQ[28]_B
AH50	LPDDR3_M0_DQ[24]_B
AH52	VSS
AH54	VSS
AH56	VSS
AH58	VSS
AH6	VSS
AH8	VSS
AK1	USB3_V1P05A_G3
AK11	VSS
AK14	VSS
AK16	RESERVED
AK18	RESERVED
AK20	UNCORE_VNN_S4
AK22	DDI_VGG_S0iX
AK24	DDI_VGG_S0iX
AK26	VSS
AK28	CORE1_VSFR_G3
AK3	USB3_RCOMP_P
AK31	CORE1_VSFR_G3
AK33	CORE_VCC1_S0iX
AK35	CORE_VCC1_S0iX
AK37	CORE_VCC1_S0iX
AK39	RESERVED
AK41	RESERVED
AK43	VSS
AK45	VSS
AK49	LPDDR3_DRAM_PWROK
AK5	USB3_RXN[2]
AK51	LPDDR3_CORE_PWROK
AK53	VSS
AK55	VSS
AK57	VSS
AK59	VSS
AK7	VSS
AK9	USB3_TXP[2]
AM10	USB3_TXN[2]

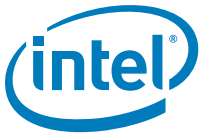


Ball #	Customer Name - LPDDR3
AM12	VSS
AM15	VSS
AM17	VSS
AM19	UNCORE_VNN_S4
AM2	VSS
AM21	DDI_VGG_S0iX
AM23	VSS
AM25	DDI_VGG_S0iX
AM27	DDI_VGG_S0iX
AM29	VSS
AM32	VSS
AM34	PWR_RSVD_OBS
AM36	DDI_VGG_S0iX
AM38	DDI_VGG_S0iX
AM4	USB3_RXP[2]
AM40	DDI_VGG_S0iX
AM42	VSS
AM44	VSS
AM46	RESERVED
AM48	LPDDR3_M1_DQ[19]_A
AM50	LPDDR3_M1_DQ[23]_A
AM52	VSS
AM54	VSS
AM56	VSS
AM58	VSS
AM6	VSS
AM8	USB3_TXP[3]
AN1	PWR_RSVD_V1P05
AN11	USB3_TXP[0]
AN3	VSS
AN49	VSS
AN5	USB3_RXP[0]
AN51	VSS
AN53	VSS
AN55	LPDDR3_M1_DQ[04]_A
AN57	LPDDR3_M1_DQ[00]_A
AN59	VSS

Ball #	Customer Name - LPDDR3
AN7	USB3_TXP[1]
AN9	USB3_TXN[3]
AP10	USB3_TXN[0]
AP12	VSS
AP14	VSS
AP16	VSS
AP18	VSS
AP2	VSS
AP20	UNCORE_VNN_S4
AP22	DDI_VGG_S0iX
AP24	DDI_VGG_S0iX
AP26	DDI_VGG_S0iX
AP28	DDI_VGG_S0iX
AP31	DDI_VGG_S0iX
AP33	DDI_VGG_S0iX
AP35	DDI_VGG_S0iX
AP37	DDI_VGG_S0iX
AP39	DDI_VGG_S0iX
AP4	USB3_RXN[0]
AP41	VSS
AP43	LPDDR3_M1_RCOMPPD
AP45	RESERVED
AP48	LPDDR3_M1_DQ[18]_A
AP50	LPDDR3_M1_DQS[2]_P_A
AP52	LPDDR3_M1_DQ[16]_A
AP54	VSS
AP56	LPDDR3_M1_DM[0]_A
AP58	VSS
AP6	USB3_TXN[1]
AP8	VSS
AR1	PWR_RSVD_V1P05
AR11	VSS
AR3	RESERVED
AR49	LPDDR3_M1_DQ[17]_A
AR5	VSS
AR51	LPDDR3_M1_DQS[2]_N_A
AR53	LPDDR3_M1_DQ[20]_A

Ball #	Customer Name - LPDDR3
AR55	LPDDR3_M1_DQ[03]_A
AR57	LPDDR3_M1_DQS[0]_P_A
AR59	DDR_VDDQG_S4
AR7	VSS
AR9	RESERVED
AT10	PCIE_TXP[1]
AT12	VSS
AT15	VSS
AT17	VSS
AT19	UNCORE_VNN_S4
AT2	RESERVED
AT21	UNCORE_VNN_S4
AT23	DDI_VGG_S0iX
AT25	VSS
AT27	VSS
AT29	DDI_VGG_S0iX
AT32	PWR_RSVD_OBS
AT34	VSS
AT36	VSS
AT38	DDI_VGG_S0iX
AT4	S_RCOMP_P
AT40	DDI_VGG_S0iX
AT42	DDRCH1_VDDQG_S4
AT44	VSS
AT46	VSS
AT48	LPDDR3_M1_DQ[22]_A
AT50	LPDDR3_M1_DQ[21]_A
AT52	VSS
AT54	VSS
AT56	VSS
AT58	LPDDR3_M1_DQS[0]_N_A
AT6	VSS
AT8	RESERVED
AU1	PCIE_V1P05_G3
AU11	PCIE_TXP[0]
AU14	F_V3P3A_G3
AU16	VSS

Ball #	Customer Name - LPDDR3
AU18	F_V1P05A_G3
AU20	VSS
AU22	UNCORE_VNN_S4
AU24	DDI_VGG_S0iX
AU26	DDI_VGG_S0iX
AU28	DDI_VGG_S0iX
AU3	S_RCOMP_N
AU31	DDI_VGG_S0iX
AU33	DDI_VGG_S0iX
AU35	DDI_VGG_S0iX
AU37	DDI_VGG_S0iX
AU39	DDI_VGG_S0iX
AU41	DDRSFRCH1_VDDQG_S4
AU43	VSS
AU45	VSS
AU49	VSS
AU5	PCIE_RXP[1]
AU51	VSS
AU53	VSS
AU55	LPDDR3_M1_DQ[01]_A
AU57	LPDDR3_M1_DQ[02]_A
AU59	DDR_VDDQG_S4
AU7	VSS
AU9	PCIE_TXN[1]
AV10	PCIE_TXN[0]
AV12	ICLK_ICOMP
AV2	VSS
AV4	PCIE_RXN[1]
AV48	LPDDR3_M1_DM[2]_A
AV50	LPDDR3_M1_DM[3]_A
AV52	LPDDR3_M1_DQ[24]_A
AV54	LPDDR3_M1_DQ[05]_A
AV56	LPDDR3_M1_DQ[06]_A
AV58	VSS
AV6	VSS
AV8	VSS
AW1	PCIE_V1P05_G3

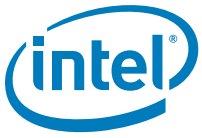


Ball #	Customer Name - LPDDR3
AW11	VSS
AW15	VSS
AW17	UNCORE1_V1P05A_G3
AW19	UNCORE1_V1P05A_G3
AW21	UNCORE_VNN_S4
AW23	DDI_VGG_S0iX
AW25	DDI_VGG_SENSE
AW27	DDI_VGG_S0iX
AW29	DDI_VGG_S0iX
AW3	PCIE_RXN[0]
AW32	DDI_VGG_S0iX
AW34	DDI_VGG_S0iX
AW36	DDI_VGG_S0iX
AW38	DDI_VGG_S0iX
AW40	DDI_VGG_S0iX
AW42	DDR_V1P05A_G3
AW44	VSS
AW46	VSS
AW49	VSS
AW5	VSS
AW51	LPDDR3_M1_DQ[29]_A
AW53	LPDDR3_M1_DQ[26]_A
AW55	LPDDR3_M1_DQ[08]_A
AW57	LPDDR3_M1_DQ[07]_A
AW59	DDR_VDDQG_S4
AW7	VSS
AW9	VSS
AY10	VSS
AY12	RESERVED
AY2	PCIE_RCOMP_P
AY4	PCIE_RXP[0]
AY48	LPDDR3_M1_DQ[27]_A
AY50	LPDDR3_M1_DQS[3]_P_A
AY52	LPDDR3_M1_DQ[25]_A
AY54	LPDDR3_M1_DQ[11]_A
AY56	VSS
AY58	LPDDR3_M1_DQ[09]_A

Ball #	Customer Name - LPDDR3
AY6	VSS
AY8	ICLK_OSCIN
B10	MIPI_V1P2A_G3
B12	MDSI_A_CLKP
B14	GPION_V1P8A_G3
B16	MCSI_2_DP[0]
B18	CORE_VCC0_S0iX
B2	VSS
B20	MCSI_1_DN[1]
B22	CORE_VCC0_S0iX
B24	CORE_VCC1_S0iX
B26	CORE_VCC1_S0iX
B28	GPIO_CAMERASB03
B32	VSS
B34	JTAG_TMS
B36	CORE_V1P15_S0iX
B38	GPIO_DFX1/C0_BPM1_TX/ C1_BPM1_TX
B4	VSS
B40	DDR_V1P05A_G3
B42	LPDDR3_M0_DQ[17]_A
B44	VSS
B46	LPDDR3_M0_DQS[2]_P_A
B48	VSS
B50	LPDDR3_M0_DQS[3]_P_A
B52	VSS
B54	LPDDR3_M0_ODT_A
B56	LPDDR3_M0_CS[1]_N
B58	VSS
B6	MDSI_C_CLKP
B8	VSS
BA1	VSS
BA11	VSS
BA14	VSS
BA16	F_V1P05A_G3
BA18	F_V1P05A_G3
BA20	UNCORE_VNN_S4
BA22	DDI_VGG_S0iX

Ball #	Customer Name - LPDDR3
BA24	DDI_VSS_SENSE
BA26	VSS
BA28	VSS
BA3	PCIE_RCOMP_N
BA31	DDI_VGG_S0iX
BA33	DDI_VGG_S0iX
BA35	VSS
BA37	VSS
BA39	DDI_VGG_S0iX
BA41	VSS
BA43	VSS
BA45	VSS
BA49	LPDDR3_M1_DQ[31]_A
BA5	PCIE_REFCLK[0]_P
BA51	LPDDR3_M1_DQS[3]_N_A
BA53	VSS
BA55	LPDDR3_M1_DQS[1]_P_A
BA57	LPDDR3_M1_DQ[10]_A
BA59	DDR_VDDQG_S4
BA7	VSS
BA9	ICLK_OSCOUT
BB10	VSS
BB12	ICLK_RCOMP
BB15	RTC_V3P3RTC_G5
BB17	F_V1P8A_G3
BB19	VSS
BB2	ICLK_VSFR_G3
BB21	UNCORE_VNN_S4
BB23	DDI_VGG_S0iX
BB25	DDI_VGG_S0iX
BB27	DDI_VGG_S0iX
BB29	DDI_VGG_S0iX
BB32	DDI_VGG_S0iX
BB34	DDI_VGG_S0iX
BB36	DDI_VGG_S0iX
BB38	DDI_VGG_S0iX
BB4	PCIE_REFCLK[0]_N

Ball #	Customer Name - LPDDR3
BB40	DDI_VGG_S0iX
BB42	VSS
BB44	VSS
BB46	VSS
BB48	LPDDR3_M1_DQ[30]_A
BB50	LPDDR3_M1_DQ[28]_A
BB52	VSS
BB54	VSS
BB56	LPDDR3_M1_DQS[1]_N_A
BB58	VSS
BB6	VSS
BB8	VSSA
BC1	ICLK_VSFR_G3
BC11	VSS
BC3	PCIE_REFCLK[1]_P
BC49	VSS
BC5	VSS
BC51	VSS
BC53	VSS
BC55	LPDDR3_M1_DM[1]_A
BC57	LPDDR3_M1_DQ[14]_A
BC59	VSS
BC7	VSS
BC9	RESERVED
BD10	RESERVED
BD12	VSS
BD14	RTC_V3P3A_G5
BD16	VSS
BD18	VSS
BD2	PCIE_REFCLK[1]_N
BD20	UNCORE_VNN_S4
BD22	DDI_VGG_S0iX
BD24	VSS
BD26	DDI_VGG_S0iX
BD28	GPIOSE_V1P8A_G3
BD31	GPIOSE_V1P8A_G3
BD33	DDI_VGG_S0iX

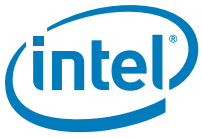


Ball #	Customer Name - LPDDR3
BD35	DDI_VGG_S0iX
BD37	DDI_VGG_S0iX
BD39	DDI_VGG_S0iX
BD4	RTC_TEST_N
BD41	VSS
BD43	VSS
BD45	VSS
BD48	LPDDR3_M1_OCAVREF
BD50	LPDDR3_M1_ODQVREF
BD52	LPDDR3_M1_CA[08]
BD54	LPDDR3_M1_DQ[15]_A
BD56	VSS
BD58	LPDDR3_M1_DQ[12]
BD6	VSS
BD8	RESERVED
BE1	VSS
BE11	VSS
BE3	PMC_RSMRST_N
BE49	RESERVED
BE5	RESERVED
BE51	RESERVED
BE53	LPDDR3_M1_CA[09]
BE55	LPDDR3_M1_DQ[13]_A
BE57	VSS
BE59	VSS
BE7	VSS
BE9	RESERVED
BF10	RTC_X2
BF12	VSS
BF15	VSS
BF17	VSS
BF19	VSS
BF2	PCIeCLK_V1P05A_G3
BF21	VSS
BF23	SDIO_V3P3A_V1P8A_G3
BF25	LPC_V3P3A_V1P8A_S4
BF27	VSS

Ball #	Customer Name - LPDDR3
BF29	GPIOSE_V1P8A_G3
BF32	VSS
BF34	VSS
BF36	VSS
BF38	PWR_RSVD_OBS
BF4	RTC_RST_N
BF40	VSS
BF42	VSS
BF44	VSS
BF46	VSS
BF48	LPDDR3_M1_CA[02]
BF50	RESERVED
BF52	VSS
BF54	LPDDR3_M1_CK_N_A
BF56	LPDDR3_M1_CK_P_A
BF58	VSS
BF6	VSS
BF8	RESERVED
BG1	PCIeCLK_V1P05A_G3
BG11	RTC_EXTPAD
BG3	PMC_CORE_PWROK
BG49	VSS
BG5	VSS
BG51	LPDDR3_M1_CA[06]
BG53	VSS
BG55	LPDDR3_M1_CK_N_B
BG57	LPDDR3_M1_CK_P_B
BG59	DDR_VDDQG_S4
BG7	VSS
BG9	RTC_X1
BH10	PMC_PWRBTN_N
BH12	VSS
BH14	MMC1_RCOMP
BH16	SD3_RCOMP
BH18	SD3_D[2]
BH2	VSS
BH20	VSS

Ball #	Customer Name - LPDDR3
BH22	LPC_RCOMP
BH24	PMC_RSTBTN_N
BH26	UART2_RTS_N
BH28	UART2_DATAIN
BH32	LPE_I2S0_CLK
BH34	I2C6_CLK/NMI_N
BH36	I2C2_DATA
BH38	VSS
BH4	PMC_BATLOW_N
BH40	LPE_I2S2_FRM
BH42	VSS
BH44	LPDDR3_M1_DQ[09]_B
BH46	LPDDR3_M1_DM[1]_B
BH48	RESERVED
BH50	LPDDR3_M1_CA[01]
BH52	VSS
BH54	LPDDR3_M1_CA[04]
BH56	VSS
BH58	VSS
BH6	PMC_SUS_STAT_N
BH8	VSS
BJ1	ICLK_VSFR_G3
BJ11	VSS
BJ13	VSS
BJ15	MMC1_CMD
BJ17	VSS
BJ19	LPC_FRAME_N/ UART0_DATAIN/SPI2_MISO
BJ21	GPIO_ALERT/ISH_GPIO[11]/ ISH_UART_DATAIN
BJ23	VSS
BJ25	FST_SPI_D[2]
BJ27	VSS
BJ3	PMC_SLP_S3_N
BJ30	LPE_I2S1_DATAIN
BJ33	NFC_I2C_CLK
BJ35	VSS
BJ37	UART0_DATAIN

Ball #	Customer Name - LPDDR3
BJ39	VSS
BJ41	VSS
BJ43	LPDDR3_M1_DQ[14]_B
BJ45	VSS
BJ47	LPDDR3_M1_DQ[07]_B
BJ49	LPDDR3_M1_CA[00]
BJ5	PMC_PLTRST_N
BJ51	LPDDR3_M1_CA[03]
BJ53	RESERVED
BJ55	LPDDR3_M1_CA[05]
BJ57	LPDDR3_M1_CA[07]
BJ59	DDR_VDDQG_S4
BJ7	PMC_WAKE_N
BJ9	PMC_SLP_S4_N
BK10	ISH_GPIO[6]/I2S4_DATAOUT
BK12	MMC1_D[3]
BK14	MMC1_D[1]
BK16	SD3_D[0]
BK18	SPI1_MOSI
BK2	VSS
BK20	LPC_CLKOUT[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT
BK22	PMC_SUSPWRDNACK
BK24	VSS
BK26	FST_SPI_D[1]
BK28	UART2_DATAOUT
BK32	LPE_I2S0_FRM
BK34	I2C5_CLK
BK36	SPI3_MOSI
BK38	LPE_I2S2_CLK
BK4	PMC_ACPRESENT
BK40	LPE_I2S2_DATAOUT
BK42	LPDDR3_M1_DQ[15]_B
BK44	LPDDR3_M1_DQ[10]_B
BK46	LPDDR3_M1_DQS[1]_P_B
BK48	VSS
BK50	VSS

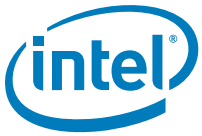


Ball #	Customer Name - LPDDR3
BK52	LPDDR3_M1_DQ[03]_B
BK54	VSS
BK56	LPDDR3_M1_CKE[0]_A
BK58	VSS
BK6	VSS
BK8	ISH_GPIO[9]/ISH_SPI_MISO/ I2S5_FS
BL1	UNCORE_VNN_S4
BL11	SD2_CLK
BL13	VSS
BL15	SD3_D[3]
BL17	SD3_CLK
BL19	SPI1_MISO
BL21	LPC_CLKRUN_N/ UART0_DATAOUT/SPI2_CLK
BL23	VSS
BL25	FST_SPI_D[3]
BL27	UART2_CTS_N
BL3	PMC_SLP_S0IX_N
BL30	VSS
BL33	I2C6_DATA/SD3_WP
BL35	I2C2_CLK
BL37	UART0_DATAOUT/SPI3_CLK
BL39	LPE_I2S2_DATAIN
BL41	VSS
BL43	LPDDR3_M1_DQ[11]_B
BL45	LPDDR3_M1_DQ[08]_B
BL47	LPDDR3_M1_DQS[1]_N_B
BL49	LPDDR3_M1_DQ[01]_B
BL5	VSS
BL51	LPDDR3_M1_DQ[00]_B
BL53	LPDDR3_M1_DQ[02]_B
BL55	RESERVED
BL57	LPDDR3_M1_CS[0]_N
BL59	DDR_VDDQG_S4
BL7	VSS
BL9	ISH_GPIO[8]/I2S5_CLK
BM10	VSS

Ball #	Customer Name - LPDDR3
BM12	MMC1_D[0]
BM14	MMC1_D[2]
BM16	VSS
BM18	VSS
BM2	ISH_GPIO[7]/I2S4_DATAIN
BM20	LPC_CLKOUT[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN
BM22	VSS
BM24	LPC_SERIRQ/SPI2_CS[0]_N
BM26	VSS
BM28	VSS
BM32	LPE_I2S0_DATAOUT
BM34	VSS
BM36	VSS
BM38	SPI3_CS[0]_N
BM4	ISH_GPIO[3]/I2S3_DATAIN
BM40	VSS
BM42	LPDDR3_M1_DQ[13]_B
BM44	VSS
BM46	VSS
BM48	LPDDR3_M1_DQ[06]_B
BM50	LPDDR3_M1_DQS[0]_P_B
BM52	LPDDR3_M1_DQ[04]_B
BM54	RESERVED
BM56	VSS
BM58	LPDDR3_M1_CKE[0]_B
BM6	PMC_SUSCLK[0]
BM8	ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT
BN1	UNCORE_VNN_S4
BN11	SD2_CMD
BN13	VSS
BN15	MMC1_CLK
BN17	VSS
BN19	SPI1_CLK
BN21	VSS
BN23	VSS

Ball #	Customer Name - LPDDR3
BN25	FST_SPI_D[0]
BN27	VSS
BN3	ISH_I2C1_CLK/ISH_SPI_CLK/ I2S5_DATAIN
BN30	VSS
BN33	NFC_I2C_DATA
BN35	VSS
BN37	SPI3_MISO
BN39	VSS
BN41	VSS
BN43	LPDDR3_M1_DQ[12]_B
BN45	VSS
BN47	LPDDR3_M1_DM[0]_B
BN49	LPDDR3_M1_DQ[05]_B
BN5	ISH_GPIO[1]/I2S3_FS
BN51	LPDDR3_M1_DQS[0]_N_B
BN53	VSS
BN55	RESERVED
BN57	RESERVED
BN59	DDR_VDDQG_S4
BN7	VSS
BN9	RESERVED
BP10	VSS
BP12	SD2_D[3]_CD_N
BP14	MMC1_D[6]
BP16	SD3_D[1]
BP18	VSS
BP2	UNCORE_VNN_S4
BP20	LPC_AD[2]/ISH_GPIO[14]
BP22	USB_OC[0]_N
BP24	FST_SPI_CLK
BP26	VSS
BP28	LPE_I2S1_DATAOUT
BP32	VSS
BP34	I2C4_CLK/DDI2_DDC_CLK
BP36	I2C1_CLK
BP38	GPIO_SW93
BP4	ISH_GPIO[5]/I2S4_FS

Ball #	Customer Name - LPDDR3
BP40	VSS
BP42	LPDDR3_M1_DQ[28]_B
BP44	LPDDR3_M1_DQ[31]_B
BP46	LPDDR3_M1_DM[3]_B
BP48	VSS
BP50	LPDDR3_M1_DQ[19]_B
BP52	LPDDR3_M1_DQ[16]_B
BP54	RESERVED
BP56	RESERVED
BP58	VSS
BP6	VSS
BP8	PMC_PLT_CLK[0]/ ISH_GPIO[10]/ ISH_UART_DATAOUT
BR1	VSS
BR11	SD2_D[1]
BR13	MMC1_D[4]
BR15	SD3_CMD
BR17	SPI1_CS[1]_N
BR19	LPC_AD[1]/ISH_GPIO[13]/ ISH_UART_RTS_N
BR21	LPC_AD[3]/ISH_GPIO[15]/ SPI2_MOSI
BR23	FST_SPI_CS[1]_N
BR25	UART1_RTS_N
BR27	UART1_CTS_N
BR3	ISH_GPIO[0]/I2S3_CLK
BR30	LPE_I2S1_CLK
BR33	I2C5_DATA
BR35	I2C1_DATA
BR37	ISH_GPIO[12]/ ISH_UART_CTS_N
BR39	PCIE_CLKREQ[0]_N
BR41	VSS
BR43	LPDDR3_M1_DQ[29]_B
BR45	LPDDR3_M1_DQ[27]_B
BR47	LPDDR3_M1_DQ[23]_B
BR49	LPDDR3_M1_DM[2]_B
BR5	VSS

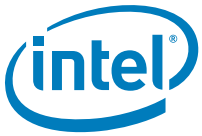


Ball #	Customer Name - LPDDR3
BR51	LPDDR3_M1_DQ[18]_B
BR53	VSS
BR55	RESERVED
BR57	RESERVED
BR59	VSS
BR7	PMC_PLT_CLK[4]/ ISH_GPIO[14]/ ISH_I2C0_DATA
BR9	PMC_PLT_CLK[1]/ ISH_GPIO[11]/ ISH_UART_DATAIN
BT10	SD2_D[0]
BT12	VSS
BT14	MMC1_D[7]
BT16	VSS
BT18	SPI1_CS[0]_N
BT2	ISH_GPIO[2]/I2S3_DATAOUT
BT20	VSS
BT22	SD3_1P8_EN
BT24	VSS
BT26	UART1_DATAIN/ UART0_DATAIN
BT28	VSS
BT32	I2C4_DATA/DDI2_DDC_DATA
BT34	VSS
BT36	I2C0_CLK
BT38	VSS
BT4	ISH_GPIO[4]/I2S4_CLK
BT40	SD3_WP
BT42	VSS
BT44	LPDDR3_M1_DQ[26]_B
BT46	VSS
BT48	LPDDR3_M1_DQ[21]_B
BT50	VSS
BT52	LPDDR3_M1_DQ[20]_B
BT54	VSS
BT56	VSS
BT58	LPDDR3_M1_CKE[1]_A

Ball #	Customer Name - LPDDR3
BT6	PMC_PLT_CLK[5]/ ISH_GPIO[15]/ISH_I2C0_CLK
BT8	VSS
BU1	VSS
BU11	SD2_D[2]
BU13	MMC1_RCLK
BU15	MMC1_D[5]
BU17	SD3_CD_N
BU19	LPC_AD[0]/ISH_GPIO[12]/ ISH_UART_CTS_N
BU21	USB_OC[1]_N
BU23	SD3_PWREN_N
BU25	FST_SPI_CS[0]_N
BU27	UART1_DATAOUT/ UART0_DATAOUT
BU3	VSS
BU30	LPE_I2S0_DATAIN
BU33	I2C3_DATA
BU35	I2C0_DATA
BU37	GPIO_SW78
BU39	PCIE_CLKREQ[1]_N
BU41	VSS
BU43	LPDDR3_M1_DQ[25]_B
BU45	LPDDR3_M1_DQ[24]_B
BU47	LPDDR3_M1_DQS[3]_N_B
BU49	LPDDR3_M1_DQ[22]_B
BU5	PWM[1]/ISH_GPIO[10]/ ISH_UART_DATAOUT
BU51	LPDDR3_M1_DQS[2]_N_B
BU53	LPDDR3_M1_DQ[17]_B
BU55	LPDDR3_M1_ODT_A
BU57	LPDDR3_M1_CKE[1]_B
BU59	VSS
BU7	PMC_PLT_CLK[2]/ ISH_GPIO[12]/ ISH_UART_CTS_N
BU9	PMC_PLT_CLK[3]/ ISH_GPIO[13]/ ISH_UART_RTS_N
BV10	DDI_VGG_S0iX

Ball #	Customer Name - LPDDR3
BV12	VSS
BV14	DDI_VGG_S0iX
BV16	VSS
BV18	LPC_V3P3A_V1P8A_S4
BV2	VSS
BV20	VSS
BV22	GPIOSE_V1P8A_G3
BV24	FST_SPI_CS[2]_N
BV26	GPIOSE_V1P8A_G3
BV28	LPE_I2S1_FRM
BV32	UNCORE_V1P8A_G3
BV34	I2C3_CLK
BV36	DDI_V1P15_S0iX
BV38	MMC1_RESET_N / SPI3_CS[1]_N
BV4	PWM[0]
BV40	DDR_V1P05A_G3
BV42	LPDDR3_M1_DQ[30]_B
BV44	VSS
BV46	LPDDR3_M1_DQS[3]_P_B
BV48	VSS
BV50	LPDDR3_M1_DQS[2]_P_B
BV52	VSS
BV54	LPDDR3_M1_ODT_B
BV56	LPDDR3_M1_CS[1]_N
BV58	VSS
BV6	VSS
BV8	VSS
BW11	DDI_VGG_S0iX
BW13	DDI_VGG_S0iX
BW15	SDIO_V3P3A_V1P8A_G3
BW17	LPC_V3P3A_V1P8A_S4
BW19	VSS
BW21	GPIOSE_V1P8A_G3
BW23	GPIOSE_V1P8A_G3
BW25	GPIOSE_V1P8A_G3
BW27	GPIOSE_V1P8A_G3

Ball #	Customer Name - LPDDR3
BW3	VSS
BW30	VSS
BW33	UNCORE_V1P8A_G3
BW35	DDI_V1P15_S0iX
BW37	DDI_V1P15_S0iX
BW39	VSS
BW41	DDR_V1P05A_G3
BW43	DDR_V1P05A_G3
BW45	VSS
BW47	VSS
BW49	VSS
BW5	VSS
BW51	VSS
BW53	VSS
BW55	VSS
BW57	VSS
BW7	VSS
BW9	DDI_VGG_S0iX
C1	VSS
C11	MDSI_A_CLKN
C13	RESERVED
C15	MCSI_2_DN[0]
C17	MCSI_1_CLKP
C19	MCSI_1_DP[1]
C21	DDI2_HPD
C23	CORE_VCC1_S0iX
C25	VSS
C27	GPIO_CAMERASB07
C3	DDI1_TXP[3]
C30	GPIO_CAMERASB04
C33	SVID_ALERT_N

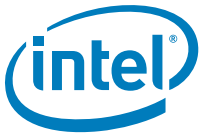


Ball #	Customer Name - LPDDR3
C35	GPIO_SUS5/PMC_SUSCLK[1]
C37	GPIO_SUS3/JTAG2_TDI
C39	GPIO_DFX2/ISH_GPIO[13]/ C0_BPM2_TX
C41	VSS
C43	LPDDR3_M0_DQ[22]_A
C45	LPDDR3_M0_DQ[23]_A
C47	LPDDR3_M0_DQS[2]_N_A
C49	LPDDR3_M0_DQ[25]_A
C5	MDSI_C_CLKN
C51	LPDDR3_M0_DQS[3]_N_A
C53	LPDDR3_M0_DQ[30]_A
C55	LPDDR3_M0_ODT_B
C57	LPDDR3_M0_CKE[1]_A
C59	VSS
C7	MDSI_C_DP[2]
C9	MDSI_C_DP[0]
D10	MDSI_A_DN[1]
D12	RESERVED
D14	RESERVED
D16	MCSI_1_CLKN
D18	MCSI_1_DN[2]
D2	DDI1_TXP[2]
D20	VSS
D22	VSS
D24	CORE_VCC1_S0iX
D26	GPIO_CAMERASB10
D28	VSS
D32	SVID_DATA
D34	VSS
D36	GPIO_SUS6/PMC_SUSCLK[2]

Ball #	Customer Name - LPDDR3
D38	VSS
D4	DDI1_TXN[3]
D40	GPIO_DFX0/C0_BPM0_TX
D42	VSS
D44	LPDDR3_M0_DQ[21]_A
D46	VSS
D48	LPDDR3_M0_DQ[26]_A
D50	VSS
D52	LPDDR3_M0_DQ[27]_A
D54	VSS
D56	VSS
D58	LPDDR3_M0_CKE[1]_B
D6	MDSI_C_DN[2]
D8	MDSI_C_DN[0]
E1	VSS
E11	VSS
E13	RESERVED
E15	VSS
E17	MCSI_1_DP[2]
E19	VSS
E21	DDI2_DDC_CLK/MDSI_A_TE/ UART0_DATAOUT
E23	CORE_VCC1_S0iX
E25	DDI1_HPD
E27	GPIO_CAMERASB06
E3	DDI1_TXN[2]
E30	GPIO_CAMERASB02
E33	JTAG_TDI
E35	JTAG_TRST_N
E37	GPIO_SUS0
E39	GPIO_DFX3/C0_BPM3_TX

Ball #	Customer Name - LPDDR3
E41	VSS
E43	LPDDR3_M0_DQ[18]_A
E45	LPDDR3_M0_DQ[20]_A
E47	LPDDR3_M0_DQ[24]_A
E49	LPDDR3_M0_DM[03]_A
E5	DDI1_TXN[1]
E51	LPDDR3_M0_DQ[29]_A
E53	VSS
E55	RESERVED
E57	RESERVED
E59	VSS
E7	VSS
E9	MDSI_A_DP[1]
F10	VSS
F12	MDSI_A_DP[2]
F14	VSS
F16	VSS
F18	VSS
F2	VSS
F20	DDI2_DDC_DATA/ MDSI_C_TE/UART0_DATAIN
F22	DDI0_VDDEN
F24	CORE_VCC1_S0iX
F26	DDI0_DDC_CLK/ DDI1_DDC_CLK
F28	GPIO_CAMERASB11
F32	SVID_CLK
F34	JTAG_TCK
F36	GPIO_SUS8
F38	GPIO_DFX7/C0_BPM2_TX
F4	DDI1_TXP[1]
F40	VSS

Ball #	Customer Name - LPDDR3
F42	LPDDR3_M0_DQ[19]_A
F44	LPDDR3_M0_DQ[16]_A
F46	LPDDR3_M0_DM[2]_A
F48	VSS
F50	LPDDR3_M0_DQ[28]_A
F52	LPDDR3_M0_DQ[31]_A
F54	RESERVED
F56	RESERVED
F58	VSS
F6	VSS
F8	VSS
G1	DDI2_VDDQ_G3
G11	MDSI_A_DN[2]
G13	VSS
G15	MCSI_2_DP[1]
G17	VSS
G19	MCSI_1_DP[3]
G21	VSS
G23	CORE_VCC1_S0iX
G25	VSS
G27	GPIO_CAMERASB05
G3	DDI1_TXN[0]
G30	VSS
G33	JTAG_PRDY_N
G35	VSS
G37	GPIO_SUS4/JTAG2_TDO
G39	GPIO_DFX5/C0_BPM0_TX/ C1_BPM0_TX
G41	VSS
G43	LPDDR3_M0_DQ[03]_A
G45	VSS
G47	LPDDR3_M0_DM[1]_A
G49	LPDDR3_M0_DQ[10]_A
G5	VSS
G51	LPDDR3_M0_DQS[1]_N_A
G53	VSS
G55	RESERVED

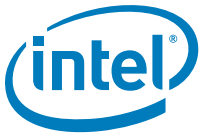


Ball #	Customer Name - LPDDR3
G57	RESERVED
G59	DDR_VDDQG_S4
G7	MDSI_C_DP[1]
G9	MDSI_A_DP[3]
H10	MDSI_A_DN[3]
H12	VSS
H14	MCSI_2_DN[1]
H16	MCSI_2_CLKP
H18	MCSI_1_DN[3]
H2	DDI1_AUXP
H20	MCSI_1_DP[0]
H22	DDI1_VDDEN/ MDSI_DDC_DATA
H24	CORE_VCC1_S0iX
H26	DDI0_DDC_DATA/ DDI1_DDC_DATA
H28	VSS
H32	PROCHOT_N
H34	JTAG_PREQ_N
H36	VSS
H38	GPIO_DFX4
H4	DDI1_TXP[0]
H40	VSS
H42	LPDDR3_M0_DQ[02]_A
H44	VSS
H46	VSS
H48	LPDDR3_M0_DQ[09]_A
H50	LPDDR3_M0_DQS[1]_P_A
H52	LPDDR3_M0_DQ[11]_A
H54	RESERVED
H56	VSS
H58	LPDDR3_M0_CKE[0]_A
H6	MDSI_C_DN[1]
H8	MDSI_C_DN[3]
J1	DDI2_VDDQ_G3
J11	MDSI_A_DP[0]
J13	VSS
J15	MCSI_2_CLKN

Ball #	Customer Name - LPDDR3
J17	MCSI_3_CLKP
J19	MCSI_1_DN[0]
J21	DDI1_BKLTEN/ MDSI_DDC_CLK
J23	CORE_VCC1_S0iX
J25	CORE_VCC1_S0iX
J27	GPIO_CAMERASB09
J3	DDI1_AUXN
J30	GPIO_CAMERASB00
J33	JTAG_TDO
J35	GPIO_SUS9
J37	GPIO_SUS2/JTAG2_TMS
J39	GPIO_DFX8/C0_BPM3_TX/ C1_BPM3_TX
J41	VSS
J43	LPDDR3_M0_DQ[04]_A
J45	LPDDR3_M0_DQ[07]_A
J47	LPDDR3_M0_DQS[0]_N_A
J49	LPDDR3_M0_DQ[14]_A
J5	DDI0_TXP[0]
J51	LPDDR3_M0_DQ[15]_A
J53	LPDDR3_M0_DQ[13]_A
J55	RESERVED
J57	LPDDR3_M0_CS[0]_N
J59	DDR_VDDQG_S4
J7	DDI1_RCOMP_N
J9	MDSI_C_DP[3]
K10	VSS
K12	MDSI_A_DN[0]
K14	MDSI_RCOMP
K16	MCSI_3_CLKN
K18	MCSI_RCOMP
K2	VSS
K20	DDI1_BKLTCTL/MDSI_A_TE/ MDSI_C_TE
K22	DDI0_BKLTCTL
K24	CORE_VCC1_S0iX
K26	DDI0_HPD

Ball #	Customer Name - LPDDR3
K28	GPIO_CAMERASB08
K32	GPIO_CAMERASB01
K34	GPIO_SUS10
K36	GPIO_SUS7/PMC_SUSCLK[3]
K38	GPIO_SUS1/JTAG2_TCK
K4	DDI0_TXN[0]
K40	GPIO_DFX6/C0_BPM1_TX/ C1_BPM1_TX
K42	LPDDR3_M0_DQ[00]_A
K44	LPDDR3_M0_DQ[05]_A
K46	LPDDR3_M0_DQS[0]_P_A
K48	VSS
K50	VSS
K52	LPDDR3_M0_DQ[12]_A
K54	VSS
K56	LPDDR3_M0_CKE[0]_B
K58	VSS
K6	VSS
K8	DDI1_RCOMP_P
L1	DDI1_VDDQ_G3
L11	VSS
L13	VSS
L15	VSS
L17	VSS
L19	VSS
L21	DDI0_BKLTEN
L23	CORE_VCC1_S0iX
L25	CORE_VCC1_S0iX
L27	VSS
L3	VSS
L30	VSS
L33	VSS
L35	VSS
L37	PWR_RSVD_OBS
L39	VSS
L41	VSS
L43	LPDDR3_M0_DQ[01]_A
L45	VSS

Ball #	Customer Name - LPDDR3
L47	LPDDR3_M0_DQ[08]_A
L49	LPDDR3_M0_CA[0]
L5	DDI0_TXN[1]
L51	LPDDR3_M0_CA[3]
L53	RESERVED
L55	LPDDR3_M0_CA[5]
L57	LPDDR3_M0_CA[7]
L59	DDR_VDDQG_S4
L7	VSS
L9	DDI0_AUXP
M10	DDI0_TXN[3]
M12	VSS
M14	VSS
M16	VSS
M18	VSS
M2	DDI1_VDDQ_G3
M20	VSS
M22	VSS
M24	CORE_VCC1_S0iX
M26	CORE_VCC1_S0iX
M28	CORE_VCC1_S0iX
M32	VSS
M34	VSS
M36	VSS
M38	VSS
M4	DDI0_TXP[1]
M40	GPIO0_RCOMP
M42	VSS
M44	LPDDR3_M0_DQ[06]_A
M46	LPDDR3_M0_DM[0]_A
M48	RESERVED
M50	LPDDR3_M0_CA[1]
M52	VSS
M54	LPDDR3_M0_CA[4]
M56	VSS
M58	VSS
M6	DDI0_TXP[2]

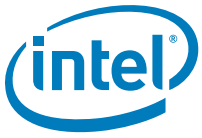


Ball #	Customer Name - LPDDR3
M8	DDI0_AUXN
N1	USB_V3P3A_G3
N11	DDI2_TXN[0]
N3	VSS
N49	VSS
N5	DDI0_TXN[2]
N51	LPDDR3_M0_CA[6]
N53	VSS
N55	LPDDR3_M0_CK_N_A
N57	LPDDR3_M0_CK_P_A
N59	DDR_VDDQG_S4
N7	VSS
N9	DDI0_TXP[3]
P10	DDI2_TXP[0]
P12	VSS
P15	VSS
P17	VSS
P19	VSS
P2	USB_V3P3A_G3
P21	VSS
P23	VSS
P25	CORE_VCC1_S0iX
P27	CORE_VCC1_S0iX
P29	VSS
P32	CORE_VCC0_S0iX
P34	CORE_VCC0_S0iX
P36	CORE_VCC0_S0iX
P38	CORE_VCC0_S0iX
P4	DDI0_RCOMP_N
P40	VSS
P42	VSS
P44	VSS
P46	VSS
P48	LPDDR3_M0_CA[2]
P50	RESERVED
P52	VSS
P54	LPDDR3_M0_CK_N_B

Ball #	Customer Name - LPDDR3
P56	LPDDR3_M0_CK_P_B
P58	VSS
P6	VSS
P8	VSS
R1	VSS
R11	VSS
R3	DDI2_AUXN
R49	RESERVED
R5	DDI0_RCOMP_P
R51	RESERVED
R53	LPDDR3_M0_CA[9]
R55	LPDDR3_M0_DQ[02]_B
R57	VSS
R59	VSS
R7	VSS
R9	VSS
T10	DDI2_TXN[2]
T12	VSS
T14	UNCORE_VSFR_G3
T16	VSS
T18	VSS
T2	DDI2_TXP[1]
T20	VSS
T22	VSS
T24	VSS
T26	VSS
T28	VSS
T31	CORE_VCC0_S0iX
T33	CORE_VCC0_S0iX
T35	CORE_VCC0_S0iX
T37	CORE_VCC0_S0iX
T39	VSS
T4	DDI2_AUXP
T41	VSS
T43	VSS
T45	VSS
T48	LPDDR3_M0_OCAVREF

Ball #	Customer Name - LPDDR3
T50	LPDDR3_M0_ODQVREF
T52	LPDDR3_M0_CA[8]
T54	LPDDR3_M0_DQ[00]_B
T56	VSS
T58	LPDDR3_M0_DQ[03]_B
T6	VSS
T8	USB_VBUSSNS
U1	USB_VDDQ_G3
U11	DDI2_TXP[2]
U3	DDI2_TXN[1]
U49	VSS
U5	VSS
U51	VSS
U53	VSS
U55	LPDDR3_M0_DM[0]_B
U57	LPDDR3_M0_DQ[01]_B
U59	VSS
U7	USB_OTG_ID
U9	RESERVED
V10	USB_DN[1]
V12	USB_DP[0]
V15	VSS
V17	VSS
V19	VSS
V2	USB_VDDQ_G3
V21	UNCORE_VNN_S4
V23	UNCORE_VNN_S4
V25	UNCORE_VNN_S4
V27	PWR_RSVD_OBS
V29	VSS
V32	CORE_VCC0_S0iX
V34	CORE_VCC0_S0iX
V36	CORE_VCC0_S0iX
V38	CORE_VSS0_SENSE
V4	DDI2_TXP[3]
V40	CORE_VCC0_SENSE
V42	VSS

Ball #	Customer Name - LPDDR3
V44	VSS
V46	VSS
V48	LPDDR3_M0_DQ[17]_B
V50	LPDDR3_M0_DQ[19]_B
V52	VSS
V54	VSS
V56	LPDDR3_M0_DQS[0]_N_B
V58	VSS
V6	VSS
V8	VSS
W1	USB_VDDQ_G3
W11	USB_DN[0]
W14	VSS
W16	VSS
W18	VSS
W20	UNCORE_VNN_S4
W22	UNCORE_VNN_S4
W24	UNCORE_V1P15_S0iX
W26	UNCORE_VNN_S4
W28	UNCORE_VNN_S4
W3	USB_DN[2]
W31	CORE0_VSFR_G3
W33	CORE_VCC0_S0iX
W35	CORE_VCC0_S0iX
W37	CORE_VCC0_S0iX
W39	F_V1P15_S0iX
W41	VSS
W43	VSS
W45	VSS
W49	LPDDR3_M0_DQ[16]_B
W5	DDI2_TXN[3]
W51	LPDDR3_M0_DQS[2]_N_B
W53	VSS
W55	LPDDR3_M0_DQS[0]_P_B
W57	LPDDR3_M0_DQ[05]_B
W59	DDR_VDDQG_S4
W7	VSS



Ball #	Customer Name - LPDDR3
W9	USB_DP[1]
Y10	VSS
Y12	VSS
Y2	VSS
Y4	USB_DP[2]
Y48	LPDDR3_M0_DQ[20]_B
Y50	LPDDR3_M0_DQS[2]_P_B
Y52	LPDDR3_M0_DQ[22]_B
Y54	LPDDR3_M0_DQ[04]_B
Y56	VSS
Y58	LPDDR3_M0_DQ[06]_B
Y6	USB_DP[3]
Y8	VSS

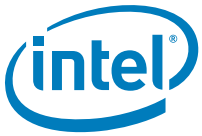
21.4 SoC Co-POP Pin List Location

Ball #	External Co-POP Naming
AY46	RTC_X1
BB46	RTC_X2
BA41	RTC_EXTPAD
BH48	PCIE_REFCLK0_N
BF48	PCIE_REFCLK1_N
BG49	PCIE_REFCLK0_P
BF46	PCIE_REFCLK1_P
AW51	PMC_CORE_PWROK
R5	JTAG_PRDY_N
T6	JTAG_PREQ_N
BC17	DDI0_AUXN
BA17	DDI0_AUXP
AP19	DDI0_RCOMP_N

Ball #	External Co-POP Naming
AT20	DDI0_RCOMP_P
BF16	DDI0_TXN0
BJ17	DDI0_TXN1
BF18	DDI0_TXN2
BF20	DDI0_TXN3
BG17	DDI0_TXP0
BH18	DDI0_TXP1
BG19	DDI0_TXP2
BG21	DDI0_TXP3
BB16	DDI1_AUXN
AY16	DDI1_AUXP
AT16	DDI1_RCOMP_N
AP17	DDI1_RCOMP_P
BG15	DDI1_TXN0

Ball #	External Co-POP Naming
BH14	DDI1_TXN1
BF12	DDI1_TXN2
BF10	DDI1_TXN3
BF14	DDI1_TXP0
BJ13	DDI1_TXP1
BG13	DDI1_TXP2
BG11	DDI1_TXP3
BE11	DDI2_AUXN
BC11	DDI2_AUXP
BH6	DDI2_TXN0
BG7	DDI2_TXN1
BK8	DDI2_TXN2
BJ9	DDI2_TXN3
BG5	DDI2_TXP0
BF8	DDI2_TXP1
BL9	DDI2_TXP2
BK10	DDI2_TXP3
E7	LPDDR3_DRAM_PWROK
G5	RESERVED
F6	RESERVED
K2	LPDDR3_M0_RCOMPPD
F10	RESERVED
G11	RESERVED
B4	LPDDR3_M1_RCOMPPD
F8	LPDDR3_CORE_PWROK
AR43	RESERVED
AU11	RESERVED
AU13	RESERVED

Ball #	External Co-POP Naming
BE41	RESERVED
E11	FST_SPI_CLK
F18	FST_SPI_CS0_N
H22	FST_SPI_CS1_N
C11	FST_SPI_CS2_N
B12	FST_SPI_D0
A7	FST_SPI_D1
E9	FST_SPI_D2
B6	FST_SPI_D3
BE47	RESERVED
BD46	RESERVED
BD42	RESERVED
AJ5	GPIO_CAMERASB00
AJ3	GPIO_CAMERASB01
AH6	GPIO_CAMERASB02
AJ13	GPIO_CAMERASB03
AG5	GPIO_CAMERASB04
AK14	GPIO_CAMERASB05
AJ11	GPIO_CAMERASB06
AJ7	GPIO_CAMERASB07
AK12	GPIO_CAMERASB08
AJ9	GPIO_CAMERASB09
AK10	GPIO_CAMERASB10
AK8	GPIO_CAMERASB11
E37	LPE_I2S2_CLK
L41	LPE_I2S2_FRM
F38	LPE_I2S2_DATAIN
A31	LPE_I2S2_DATAOUT

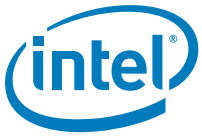


Ball #	External Co-POP Naming
D50	GPIO_ALERT/ ISH_GPIO11/ ISH_UART_DATAIN
AB12	GPIO_DFX0/ C0_BPM0_TX
Y6	GPIO_DFX1/ C0_BPM1_TX/ C1_BPM1_TX
AB10	GPIO_DFX2/ ISH_GPIO13/ C0_BPM2_TX
U5	GPIO_DFX3/ C0_BPM3_TX
U3	GPIO_DFX4
AB8	GPIO_DFX5/ C0_BPM0_TX/ C1_BPM0_TX
AA5	GPIO_DFX6/ C0_BPM1_TX/ C1_BPM1_TX
V6	GPIO_DFX7/ C0_BPM2_TX
AB4	GPIO_DFX8/ C0_BPM3_TX/ C1_BPM3_TX
AC5	GPIO_SUS0
AD10	GPIO_SUS1/JTAG2_TCK
AD12	GPIO_SUS2/JTAG2_TMS
AD2	GPIO_SUS3/JTAG2_TDI
AB6	GPIO_SUS4/JTAG2_TDO
AD6	GPIO_SUS5/ PMC_SUSCLK1
AD8	GPIO_SUS6/ PMC_SUSCLK2
AF6	GPIO_SUS7/ PMC_SUSCLK3
AA1	GPIO0_RCOMP
BC41	RESERVED
AD21	RESERVED

Ball #	External Co-POP Naming
L9	DDI0_DDC_CLK/ DDI1_DDC_CLK
K6	DDI0_DDC_DATA/ DDI1_DDC_DATA
L5	DDI0_HPD
H6	DDI1_HPD
L11	DDI2_DDC_CLK/ MDSI_A_TE/ UART0_DATAOUT
N17	DDI2_DDC_DATA/ MDSI_C_TE/ UART0_DATAIN
J5	DDI2_HPD
H28	NFC_I2C_CLK
H30	NFC_I2C_DATA
H36	I2C0_CLK
D30	I2C0_DATA
C27	I2C1_CLK
E29	I2C1_DATA
E27	I2C2_CLK
C23	I2C2_DATA
E25	I2C3_CLK
E23	I2C3_DATA
J35	I2C4_CLK/ DDI2_DDC_CLK
K36	I2C4_DATA/ DDI2_DDC_DATA
F24	I2C5_CLK
F30	I2C5_DATA
F28	I2C6_CLK/NMI_N
F26	I2C6_DATA/SD3_WP
BH50	ICLK_ICOMP
BD50	ICLK_RCOMP

Ball #	External Co-POP Naming
D46	LPC_SERIRQ/ SPI2_CS0_N
BA47	RESERVED
K28	LPC_CLKRUN_N/ UART0_DATAOUT/ SPI2_CLK
K22	LPC_FRAME_N/ UART0_DATAIN/ SPI2_MISO
C5	LPC_RCOMP
D4	RESERVED
C3	RESERVED
AN3	MCSI_1_CLKN
AN5	MCSI_1_CLKP
AK4	MCSI_1_DN0
AM6	MCSI_1_DN1
AP6	MCSI_1_DN2
AR9	MCSI_1_DN3
AK6	MCSI_1_DP0
AL5	MCSI_1_DP1
AR5	MCSI_1_DP2
AR7	MCSI_1_DP3
AU5	MCSI_2_CLKN
AT6	MCSI_2_CLKP
AT8	MCSI_2_DN0
AU3	MCSI_2_DN1
AT10	MCSI_2_DP0
AU1	MCSI_2_DP1
AT14	MCSI_3_CLKN
AT12	MCSI_3_CLKP
AV2	MCSI_RCOMP

Ball #	External Co-POP Naming
BB6	MDSI_A_CLKN
BA5	MDSI_A_CLKP
AY6	MDSI_A_DN0
BA3	MDSI_A_DN1
BD6	MDSI_A_DN2
BE3	MDSI_A_DN3
AW5	MDSI_A_DP0
AY4	MDSI_A_DP1
BC5	MDSI_A_DP2
BF4	MDSI_A_DP3
BH4	MDSI_C_CLKN
BG3	MDSI_C_CLKP
BA11	MDSI_C_DN0
BF2	MDSI_C_DN1
BJ3	MDSI_C_DN2
BK4	MDSI_C_DN3
BA9	MDSI_C_DP0
BE1	MDSI_C_DP1
BH2	MDSI_C_DP2
BJ5	MDSI_C_DP3
BK6	MDSI_RCOMP
E43	LPE_I2S0_DATAOUT
F44	LPE_I2S1_DATAIN
B44	LPE_I2S1_DATAOUT
H42	LPE_I2S0_CLK
E45	LPE_I2S1_CLK
G43	LPE_I2S1_FRM
C43	LPE_I2S0_DATAIN

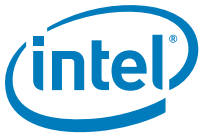


Ball #	External Co-POP Naming
L35	LPE_I2S0_FRM
AR49	ISH_GPIO0/I2S3_CLK
AV46	ISH_GPIO1/I2S3_FS
AP46	ISH_GPIO2/ I2S3_DATAOUT
AV48	ISH_GPIO3/ I2S3_DATAIN
AU47	ISH_GPIO4/I2S4_CLK
AT46	ISH_GPIO5/I2S4_FS
AN47	ISH_GPIO6/ I2S4_DATAOUT
AL47	ISH_GPIO7/ I2S4_DATAIN
AL51	ISH_GPIO8/I2S5_CLK
AL49	ISH_GPIO9/ ISH_SPI_MISO/I2S5_FS
AR47	ISH_I2C1_CLK/ ISH_SPI_CLK/ I2S5_DATAIN
AM46	ISH_I2C1_DATA/ ISH_SPI_MOSI/ I2S5_DATAOUT
N23	LPC_AD0/ISH_GPIO12/ ISH_UART_CTS_N
P28	LPC_AD1/ISH_GPIO13/ ISH_UART_RTS_N
L23	LPC_AD2/ISH_GPIO14
M22	LPC_AD3/ISH_GPIO15/ SPI2_MOSI
J23	LPC_CLKOUT0/ ISH_GPIO10/ ISH_UART_DATAOUT
M28	LPC_CLKOUT1/ ISH_GPIO11/ ISH_UART_DATAIN
BA45	PMC_PLT_CLK0/ ISH_GPIO10/ ISH_UART_DATAOUT
BA43	PMC_PLT_CLK1/ ISH_GPIO11/ ISH_UART_DATAIN

Ball #	External Co-POP Naming
AT42	PMC_PLT_CLK2/ ISH_GPIO12/ ISH_UART_CTS_N
AR41	PMC_PLT_CLK3/ ISH_GPIO13/ ISH_UART_RTS_N
AT40	PMC_PLT_CLK4/ ISH_GPIO14/ ISH_I2C0_DATA
AT44	PMC_PLT_CLK5/ ISH_GPIO15/ ISH_I2C0_CLK
C31	UART0_DATAOUT/ SPI3_CLK
F32	SPI3_MISO
E35	SPI3_MOSI
L45	MMC1_D4
G47	MMC1_D5
G49	MMC1_D6
F50	MMC1_D7
E49	MMC1_RCLK
BK48	ICLK_OSCIN
BJ49	ICLK_OSCOUT
L17	DDI0_BKLTCTL
M16	DDI0_BKLTEN
K16	DDI0_VDDEN
J17	DDI1_BKLTCTL/ MDSI_A_TE/MDSI_C_TE
F16	DDI1_BKLTEN/ MDSI_DDC_CLK
H16	DDI1_VDDEN/ MDSI_DDC_DATA
C39	PCIE_CLKREQ0_N
J41	PCIE_CLKREQ1_N
E39	GPIO_SW93
D38	SD3_WP

Ball #	External Co-POP Naming
AT27	PCIE_RCOMP_N
AP27	PCIE_RCOMP_P
BK46	PCIE_RXN0
BH46	PCIE_RXN1
BL45	PCIE_RXP0
BJ45	PCIE_RXP1
BG43	PCIE_TXN0
BG41	PCIE_TXN1
BF44	PCIE_TXP0
BF42	PCIE_TXP1
AC47	PMC_ACPRESENT
AB48	PMC_BATLOW_N
AG47	PMC_PLTRST_N
AH46	PMC_PWRBTN_N
F46	PMC_RSTBTN_N
AC51	PMC_SLP_S0IX_N
AG49	PMC_SLP_S3_N
AK46	PMC_SLP_S4_N
AC49	PMC_SUSCLK0
AK48	PMC_WAKE_N
U13	PROCHOT_N
AD46	PWM0
AC45	PWM1/ISH_GPIO10/ ISH_UART_DATAOUT
BC49	PMC_RSMRST_N
AW49	RTC_TEST_N
E33	ISH_GPIO12/ ISH_UART_CTS_N
E31	SPI3_CS[0]_N
C35	GPIO_SW78

Ball #	External Co-POP Naming
F36	MMC1_RESET_N / SPI3_CS[1]_N
F34	UART0_DATAIN
AT31	S_RCOMP_N
AP30	S_RCOMP_P
BH38	RESERVED
BF38	RESERVED
BG39	RESERVED
BF40	RESERVED
P46	MMC1_CLK
AA47	MMC1_CMD
M46	MMC1_D0
W49	MMC1_D1
N47	MMC1_D2
W47	MMC1_D3
R51	MMC1_RCOMP
M50	SD2_CLK
R49	SD2_CMD
L47	SD2_D0
L49	SD2_D1
R47	SD2_D2
J47	SD2_D3_CD_N
D48	SD3_1P8_EN
C49	SD3_CD_N
Y46	SD3_CLK
T46	SD3_CMD
H46	SD3_D0
U47	SD3_D1
V46	SD3_D2

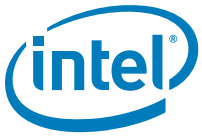


Ball #	External Co-POP Naming
AB46	SD3_D3
C47	SD3_PWREN_N
J51	SD3_RCOMP
N5	GPIO_SUS10
AE5	GPIO_SUS8
AE3	GPIO_SUS9
BJ47	RESERVED
AD25	RESERVED
AC24	RESERVED
AK40	SPI1_CLK
AH40	SPI1_CS0_N
AH42	SPI1_CS1_N
AK42	SPI1_MISO
AK44	SPI1_MOSI
BC47	RTC_RST_N
AJ47	PMC_SUS_STAT_N
B48	PMC_SUSPWRDNACK
T10	SVID_ALERT_N
N3	SVID_CLK
T12	SVID_DATA
P4	JTAG_TCK
T8	JTAG_TDI
U9	JTAG_TDO
AV24	RESERVED
AV22	RESERVED
U11	JTAG_TMS
M6	JTAG_TRST_N
C19	UART1_CTS_N

Ball #	External Co-POP Naming
E21	UART1_RTS_N
D22	UART1_DATAIN/ UART0_DATAIN
A23	UART1_DATAOUT/ UART0_DATAOUT
F20	UART2_CTS_N
F22	UART2_RTS_N
E19	UART2_DATAIN
B20	UART2_DATAOUT
BH30	USB_SSIC1_RX_N
BJ29	USB_SSIC1_RX_P
BF28	USB_SSIC1_TX_N
BG27	USB_SSIC1_TX_P
BH22	USB_DN0
BG23	USB_DN1
BJ25	USB_DN2
BA29	USB_DN3
BF22	USB_DP0
BH24	USB_DP1
BG25	USB_DP2
BC29	USB_DP3
BD24	USB_HSIC_0_DATA
BB24	USB_HSIC_0_STROBE
AY22	USB_HSIC_1_DATA
BB22	USB_HSIC_1_STROBE
BK26	USB_HSIC_RCOMP
BF24	RESERVED
M30	USB_OC0_N
K30	USB_OC1_N
AY24	USB_OTG_ID

Ball #	External Co-POP Naming
BL25	USB_RCOMP
AV28	RESERVED
AV26	RESERVED
BK36	USB_SSIC_RCOMP_N
BL35	USB_SSIC_RCOMP_P
BG31	USB_SSIC0_RX_N
BH32	USB_SSIC0_RX_P
BF30	USB_SSIC0_TX_N
BG29	USB_SSIC0_TX_P
BD22	USB_VBUSSNS
AT24	USB3_RCOMP_N
AP23	USB3_RCOMP_P
AY36	USB3_RXN0
BE35	USB3_RXN1
BA35	USB3_RXN2
BB30	USB3_RXN3
BB36	USB3_RXP0
BC35	USB3_RXP1
AW35	USB3_RXP2
BD30	USB3_RXP3
BJ37	USB3_TXN0
BG37	USB3_TXN1
BG35	USB3_TXN2
BF32	USB3_TXN3
BK38	USB3_TXP0
BF36	USB3_TXP1
BF34	USB3_TXP2
BG33	USB3_TXP3

Ball #	External Co-POP Naming
BH34	PCIE_V1P05_G3
BK34	PCIE_V1P05_G3
BJ33	PWR_RSVD_V1P05
BK32	PWR_RSVD_V1P05
BL33	PWR_RSVD_V1P05
BJ31	USB3_V1P05A_G3
BL31	USB3_V1P05A_G3
AB14	CORE_VCC0_SENSE
AD14	CORE_VSS0_SENSE
W26	CORE_VCC1_SENSE
V27	CORE_VSS1_SENSE
AP32	F_V1P05A_G3
AT35	F_V1P8A_G3
AV36	F_V3P3A_G3
BK16	DDI1_VDDQ_G3
BL15	DDI1_VDDQ_G3
BH16	DDI2_VDDQ_G3
BJ15	DDI2_VDDQ_G3
BK14	DDI2_VDDQ_G3
A19	DDR_V1P05A_G3
A21	DDR_V1P05A_G3
A37	DDR_V1P05A_G3
B36	DDR_V1P05A_G3
B38	DDR_V1P05A_G3
C21	DDR_V1P05A_G3
C37	DDR_V1P05A_G3
D20	DDR_V1P05A_G3
D36	DDR_V1P05A_G3

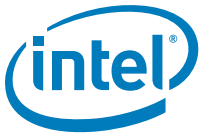


Ball #	External Co-POP Naming
R11	PWR_RSVD_OBS
AA51	UNCORE_V1P8A_G3
P22	DDRCH0_VDDQG_S4
P30	DDRCH1_VDDQG_S4
T26	PWR_RSVD_OBS
AD4	CORE0_VSFR_G3
AC1	CORE1_VSFR_G3
AC3	CORE1_VSFR_G3
A13	CORE_VCC0_S0iX
B14	CORE_VCC0_S0iX
C13	CORE_VCC0_S0iX
D12	CORE_VCC0_S0iX
D14	CORE_VCC0_S0iX
E13	CORE_VCC0_S0iX
F12	CORE_VCC0_S0iX
G13	CORE_VCC0_S0iX
H12	CORE_VCC0_S0iX
J13	CORE_VCC0_S0iX
K12	CORE_VCC0_S0iX
L13	CORE_VCC0_S0iX
M12	CORE_VCC0_S0iX
N13	CORE_VCC0_S0iX
P12	CORE_VCC0_S0iX
R13	CORE_VCC0_S0iX
T14	CORE_VCC0_S0iX
V14	CORE_VCC0_S0iX
W16	CORE_VCC0_S0iX
W18	CORE_VCC0_S0iX

Ball #	External Co-POP Naming
AA17	CORE_VCC0_S0iX
AC16	CORE_VCC0_S0iX
AC18	CORE_VCC0_S0iX
AD17	CORE_VCC0_S0iX
A15	CORE_VCC1_S0iX
A17	CORE_VCC1_S0iX
B16	CORE_VCC1_S0iX
C15	CORE_VCC1_S0iX
C17	CORE_VCC1_S0iX
D16	CORE_VCC1_S0iX
E15	CORE_VCC1_S0iX
F14	CORE_VCC1_S0iX
G15	CORE_VCC1_S0iX
H14	CORE_VCC1_S0iX
J15	CORE_VCC1_S0iX
K14	CORE_VCC1_S0iX
L15	CORE_VCC1_S0iX
M14	CORE_VCC1_S0iX
N15	CORE_VCC1_S0iX
P14	CORE_VCC1_S0iX
P16	CORE_VCC1_S0iX
T18	CORE_VCC1_S0iX
V19	CORE_VCC1_S0iX
W20	CORE_VCC1_S0iX
W22	CORE_VCC1_S0iX
W24	CORE_VCC1_S0iX
AA21	CORE_VCC1_S0iX
AA23	CORE_VCC1_S0iX

Ball #	External Co-POP Naming
AC22	CORE_VCC1_S0iX
A25	DDR_VDDQG_S4
A27	DDR_VDDQG_S4
A29	DDR_VDDQG_S4
B26	DDR_VDDQG_S4
B30	DDR_VDDQG_S4
C25	DDR_VDDQG_S4
C29	DDR_VDDQG_S4
D26	DDR_VDDQG_S4
D28	DDR_VDDQG_S4
N11	PWR_RSVD_OBS
BJ39	PCIeCLK_V1P05A_G3
BL39	PCIeCLK_V1P05A_G3
BB2	MIPI_V1P2A_G3
BB4	MIPI_V1P2A_G3
BK12	UNCORE_VSFR_G3
AP34	F_V1P05A_G3
AT33	F_V1P05A_G3
V17	F_V1P15_S0iX
V21	F_V1P15_S0iX
A41	DDI_VGG_S0iX
A43	DDI_VGG_S0iX
B40	DDI_VGG_S0iX
B42	DDI_VGG_S0iX
C41	DDI_VGG_S0iX
D40	DDI_VGG_S0iX
D42	DDI_VGG_S0iX
E41	DDI_VGG_S0iX

Ball #	External Co-POP Naming
F40	DDI_VGG_S0iX
F42	DDI_VGG_S0iX
G39	DDI_VGG_S0iX
G41	DDI_VGG_S0iX
H38	DDI_VGG_S0iX
H40	DDI_VGG_S0iX
J39	DDI_VGG_S0iX
K38	DDI_VGG_S0iX
K40	DDI_VGG_S0iX
L39	DDI_VGG_S0iX
M38	DDI_VGG_S0iX
M40	DDI_VGG_S0iX
N37	DDI_VGG_S0iX
N39	DDI_VGG_S0iX
P36	DDI_VGG_S0iX
P38	DDI_VGG_S0iX
P40	DDI_VGG_S0iX
R39	DDI_VGG_S0iX
T33	DDI_VGG_S0iX
T35	DDI_VGG_S0iX
T38	DDI_VGG_S0iX
V28	DDI_VGG_S0iX
V30	DDI_VGG_S0iX
V32	DDI_VGG_S0iX
V34	DDI_VGG_S0iX
V36	DDI_VGG_S0iX
W27	DDI_VGG_S0iX
W29	DDI_VGG_S0iX

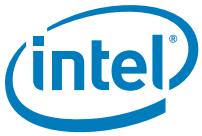


Ball #	External Co-POP Naming
W31	DDI_VGG_S0iX
W33	DDI_VGG_S0iX
AA28	DDI_VGG_S0iX
AA32	DDI_VGG_S0iX
AA34	DDI_VGG_S0iX
AC27	DDI_VGG_S0iX
AC29	DDI_VGG_S0iX
AC33	DDI_VGG_S0iX
AD30	DDI_VGG_S0iX
AD32	DDI_VGG_S0iX
AD34	DDI_VGG_S0iX
AF31	DDI_VGG_S0iX
AF33	DDI_VGG_S0iX
AH30	DDI_VGG_S0iX
AH32	DDI_VGG_S0iX
AH34	DDI_VGG_S0iX
AB2	UNCORE_VSFR_G3
AM50	LPC_V3P3A_V1P8A_S4
AH48	GPIOSE_V1P8A_G3
AH50	GPIOSE_V1P8A_G3
AJ49	GPIOSE_V1P8A_G3
AJ51	GPIOSE_V1P8A_G3
AW1	GPION_V1P8A_G3
AW3	GPION_V1P8A_G3
AY2	GPION_V1P8A_G3
BA1	GPION_V1P8A_G3
AN51	SDIO_V3P3A_V1P8A_G3
BJ27	USBHSIC_V1P2A_G3

Ball #	External Co-POP Naming
BK30	USBSSIC_V1P05A_G3
BL27	USBSSIC_V1P2A_G3
N1	CORE_V1P15_S0iX
P2	CORE_V1P15_S0iX
R1	CORE_V1P15_S0iX
R3	CORE_V1P15_S0iX
T4	CORE_V1P15_S0iX
AR51	RTC_V3P3RTC_G5
AP50	RTC_V3P3A_G5
BL37	ICLK_VSFR_G3
BJ43	ICLK_VSFR_G3
BL43	ICLK_VSFR_G3
T20	DDRSFRCH0_VDDQG_S4
N35	DDRSFRCH1_VDDQG_S4
BL21	USB_VDDQ_G3
AD19	UNCORE_VNN_S4
AE39	UNCORE_VNN_S4
AE41	UNCORE_VNN_S4
AE43	UNCORE_VNN_S4
AE45	UNCORE_VNN_S4
AE47	UNCORE_VNN_S4
AE49	UNCORE_VNN_S4
AE51	UNCORE_VNN_S4
AF20	UNCORE_VNN_S4
AF35	UNCORE_VNN_S4
AF38	UNCORE_VNN_S4
AF40	UNCORE_VNN_S4
AF42	UNCORE_VNN_S4

Ball #	External Co-POP Naming
AF44	UNCORE_VNN_S4
AF46	UNCORE_VNN_S4
AF48	UNCORE_VNN_S4
AF50	UNCORE_VNN_S4
AG39	UNCORE_VNN_S4
AG41	UNCORE_VNN_S4
AG43	UNCORE_VNN_S4
AH19	UNCORE_VNN_S4
AH21	UNCORE_VNN_S4
AH23	UNCORE_VNN_S4
AH25	UNCORE_VNN_S4
AH36	UNCORE_VNN_S4
AJ20	UNCORE_VNN_S4
AJ26	UNCORE_VNN_S4
AJ27	UNCORE_VNN_S4
AJ29	UNCORE_VNN_S4
AJ31	UNCORE_VNN_S4
AJ33	UNCORE_VNN_S4
AJ35	UNCORE_VNN_S4
AL19	UNCORE_VNN_S4
AL21	UNCORE_VNN_S4
AL23	UNCORE_VNN_S4
AL25	UNCORE_VNN_S4
M48	DDI_V1P15_S0iX
N49	DDI_V1P15_S0iX
N51	DDI_V1P15_S0iX
P48	DDI_V1P15_S0iX
P50	DDI_V1P15_S0iX

Ball #	External Co-POP Naming
Y38	PWR_RSVD_OBS
AF14	PWR_RSVD_OBS
AJ16	UNCORE_V1P15_S0iX
AM2	UNCORE_V1P15_S0iX
AM4	UNCORE_V1P15_S0iX
BH44	UNCORE1_V1P05A_G3
BK44	UNCORE1_V1P05A_G3
AL1	UNCORE2_V1P05A_G3
AL3	UNCORE2_V1P05A_G3
BL23	UNCORE_VSFR_G3
BK18	USB_V1P8A_G3
BH20	USB_V3P3A_G3
BK20	USB_V3P3A_G3
BJ23	USB_VDDQ_G3
BK22	USB_VDDQ_G3
U1	VDD1
V2	VDD1
V4	VDD1
W1	VDD1
W3	VDD1
BA49	VDD1
BA51	VDD1
BB48	VDD1
BB50	VDD1
BC51	VDD1
A45	VDD2
B46	VDD2
C45	VDD2

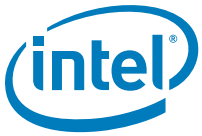


Ball #	External Co-POP Naming
D44	VDD2
F2	VDD2
G1	VDD2
G3	VDD2
H2	VDD2
H4	VDD2
BC1	VDD2
BC3	VDD2
BD2	VDD2
BD4	VDD2
BD48	VDD2
BE49	VDD2
BE51	VDD2
BF50	VDD2
A11	VDDQ
A33	VDDQ
A35	VDDQ
B10	VDDQ
B34	VDDQ
C33	VDDQ
D10	VDDQ
D34	VDDQ
AN1	VDDQ
AP2	VDDQ
AR3	VDDQ
AT48	VDDQ
AT50	VDDQ
AU49	VDDQ

Ball #	External Co-POP Naming
AU51	VDDQ
AV50	VDDQ
AH38	VGG_SENSE
AK38	DDI_VSS_SENSE
AF22	UNCORE_VNN_SENSE
AF24	UNCORE_VSS_SENSE
A1	VSS
A3	VSS
A5	VSS
A9	VSS
A39	VSS
A47	VSS
A49	VSS
A51	VSS
B2	VSS
B8	VSS
B18	VSS
B22	VSS
B24	VSS
B28	VSS
B32	VSS
B50	VSS
C1	VSS
C7	VSS
C9	VSS
C51	VSS
D2	VSS
D6	VSS

Ball #	External Co-POP Naming
D8	VSS
D18	VSS
D24	VSS
D32	VSS
E1	VSS
E3	VSS
E5	VSS
E17	VSS
E47	VSS
E51	VSS
F4	VSS
F48	VSS
G7	VSS
G9	VSS
G17	VSS
G21	VSS
G23	VSS
G27	VSS
G29	VSS
G33	VSS
G35	VSS
G37	VSS
G45	VSS
G51	VSS
H8	VSS
H10	VSS
H18	VSS
H20	VSS

Ball #	External Co-POP Naming
H24	VSS
H34	VSS
H44	VSS
H48	VSS
H50	VSS
J1	VSS
J3	VSS
J7	VSS
J9	VSS
J11	VSS
J21	VSS
J27	VSS
J43	VSS
J45	VSS
J49	VSS
K4	VSS
K8	VSS
K10	VSS
K24	VSS
K44	VSS
K46	VSS
K48	VSS
K50	VSS
L1	VSS
L3	VSS
L7	VSS
L19	VSS
L21	VSS

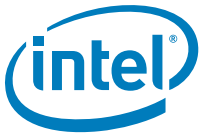


Ball #	External Co-POP Naming
L25	VSS
L29	VSS
L33	VSS
L43	VSS
L51	VSS
M2	VSS
M4	VSS
M8	VSS
M10	VSS
M20	VSS
M26	VSS
N7	VSS
N9	VSS
P6	VSS
P8	VSS
P10	VSS
R7	VSS
R9	VSS
R43	VSS
R45	VSS
T2	VSS
T22	VSS
T24	VSS
T31	VSS
T44	VSS
T48	VSS
T50	VSS
U7	VSS

Ball #	External Co-POP Naming
U43	VSS
U45	VSS
U49	VSS
U51	VSS
V12	VSS
V23	VSS
V25	VSS
V42	VSS
V44	VSS
V48	VSS
V50	VSS
W5	VSS
W7	VSS
W9	VSS
W11	VSS
W35	VSS
W41	VSS
W43	VSS
W45	VSS
W51	VSS
Y2	VSS
Y4	VSS
Y8	VSS
Y10	VSS
Y40	VSS
Y42	VSS
Y44	VSS
Y48	VSS

Ball #	External Co-POP Naming
Y50	VSS
AA3	VSS
AA7	VSS
AA9	VSS
AA11	VSS
AA19	VSS
AA27	VSS
AA30	VSS
AA36	VSS
AA49	VSS
AB50	VSS
AC13	VSS
AC20	VSS
AC26	VSS
AC31	VSS
AC35	VSS
AC39	VSS
AC41	VSS
AC43	VSS
AD23	VSS
AD27	VSS
AD28	VSS
AD36	VSS
AD38	VSS
AD40	VSS
AD42	VSS
AD44	VSS
AD48	VSS

Ball #	External Co-POP Naming
AD50	VSS
AE1	VSS
AE13	VSS
AF2	VSS
AF4	VSS
AF18	VSS
AF26	VSS
AF27	VSS
AF29	VSS
AG1	VSS
AG3	VSS
AG51	VSS
AH4	VSS
AH12	VSS
AH17	VSS
AH27	VSS
AH28	VSS
AJ1	VSS
AJ18	VSS
AJ22	VSS
AJ24	VSS
AJ41	VSS
AK2	VSS
AK50	VSS
AL7	VSS
AL9	VSS
AL13	VSS
AL17	VSS

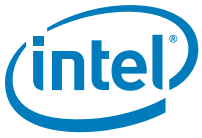


Ball #	External Co-POP Naming
AL27	VSS
AL28	VSS
AL30	VSS
AL32	VSS
AL34	VSS
AL36	VSS
AL39	VSS
AL41	VSS
AM8	VSS
AM14	VSS
AM38	VSS
AM40	VSS
AM48	VSS
AN7	VSS
AN13	VSS
AN16	VSS
AN18	VSS
AN20	VSS
AN24	VSS
AN33	VSS
AN35	VSS
AN41	VSS
AN45	VSS
AN49	VSS
AP4	VSS
AP21	VSS
AP25	VSS
AP28	VSS

Ball #	External Co-POP Naming
AP36	VSS
AP38	VSS
AP44	VSS
AP48	VSS
AR11	VSS
AR13	VSS
AR39	VSS
AR45	VSS
AT2	VSS
AT4	VSS
AT18	VSS
AT22	VSS
AT26	VSS
AT29	VSS
AU7	VSS
AU39	VSS
AU41	VSS
AU45	VSS
AV4	VSS
AV6	VSS
AV14	VSS
AV16	VSS
AV18	VSS
AV20	VSS
AV30	VSS
AV32	VSS
AV34	VSS
AV38	VSS

Ball #	External Co-POP Naming
AV44	VSS
AW13	VSS
AW17	VSS
AW19	VSS
AW21	VSS
AW23	VSS
AW29	VSS
AW31	VSS
AW33	VSS
AW43	VSS
AW45	VSS
AW47	VSS
AY12	VSS
AY14	VSS
AY18	VSS
AY20	VSS
AY30	VSS
AY38	VSS
AY40	VSS
AY42	VSS
AY44	VSS
AY48	VSS
AY50	VSS
BA13	VSS
BA31	VSS
BA37	VSS
BB8	VSS
BB10	VSS

Ball #	External Co-POP Naming
BB12	VSS
BB14	VSS
BB32	VSS
BB34	VSS
BB42	VSS
BB44	VSS
BC7	VSS
BC9	VSS
BC13	VSS
BC25	VSS
BC27	VSS
BC33	VSS
BC43	VSS
BC45	VSS
BD8	VSS
BD10	VSS
BD12	VSS
BD14	VSS
BD16	VSS
BD32	VSS
BD34	VSS
BD36	VSS
BD40	VSS
BD44	VSS
BE5	VSS
BE7	VSS
BE9	VSS
BE13	VSS

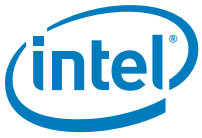


Ball #	External Co-POP Naming
BE23	VSS
BE29	VSS
BE31	VSS
BE33	VSS
BE37	VSS
BE39	VSS
BE43	VSS
BE45	VSS
BF6	VSS
BF26	VSS
BG1	VSS
BG9	VSS
BG45	VSS
BG47	VSS
BG51	VSS
BH8	VSS
BH10	VSS
BH12	VSS
BH26	VSS
BH28	VSS
BH36	VSS
BH40	VSS
BH42	VSS
BJ1	VSS
BJ7	VSS
BJ11	VSS
BJ19	VSS
BJ21	VSS

Ball #	External Co-POP Naming
BJ35	VSS
BJ41	VSS
BJ51	VSS
BK2	VSS
BK24	VSS
BK28	VSS
BK42	VSS
BK50	VSS
BL1	VSS
BL3	VSS
BL5	VSS
BL7	VSS
BL11	VSS
BL13	VSS
BL17	VSS
BL19	VSS
BL29	VSS
BL47	VSS
BL49	VSS
BL51	VSS
G19	VSS
G25	VSS
G31	VSS
H26	VSS
H32	VSS
J19	VSS
J25	VSS
J29	VSS

Ball #	External Co-POP Naming
J31	VSS
J33	VSS
J37	VSS
K18	VSS
K20	VSS
K26	VSS
K32	VSS
K34	VSS
K42	VSS
L27	VSS
L31	VSS
L37	VSS
M18	VSS
M24	VSS
M32	VSS
M34	VSS
M36	VSS
M42	VSS
M44	VSS
N19	VSS
N21	VSS
N25	VSS
N27	VSS
N29	VSS
N31	VSS
N33	VSS
N41	VSS
N43	VSS

Ball #	External Co-POP Naming
N45	VSS
P18	VSS
P20	VSS
P24	VSS
P26	VSS
P32	VSS
P34	VSS
P42	VSS
P44	VSS
R41	VSS
T27	VSS
T29	VSS
T40	VSS
T42	VSS
U39	VSS
U41	VSS
V8	VSS
V10	VSS
V38	VSS
V40	VSS
W13	VSS
W39	VSS
Y12	VSS
Y14	VSS
AA13	VSS
AA25	VSS
AA39	VSS
AA41	VSS



Ball #	External Co-POP Naming
AA43	VSS
AA45	VSS
AB38	VSS
AB40	VSS
AB42	VSS
AB44	VSS
AC7	VSS
AC9	VSS
AC11	VSS
AE7	VSS
AE9	VSS
AE11	VSS
AF8	VSS
AF10	VSS
AF12	VSS
AF16	VSS
AG7	VSS
AG9	VSS
AG11	VSS
AG13	VSS
AG45	VSS
AH8	VSS
AH10	VSS
AH14	VSS
AH44	VSS
AJ39	VSS
AJ43	VSS
AJ45	VSS

Ball #	External Co-POP Naming
AL11	VSS
AL43	VSS
AL45	VSS
AM10	VSS
AM12	VSS
AM42	VSS
AM44	VSS
AN9	VSS
AN11	VSS
AN22	VSS
AN26	VSS
AN27	VSS
AN29	VSS
AN31	VSS
AN39	VSS
AN43	VSS
AP8	VSS
AP10	VSS
AP12	VSS
AP14	VSS
AP40	VSS
AP42	VSS
AT38	VSS
AU9	VSS
AU43	VSS
AV8	VSS
AV10	VSS
AV12	VSS

Ball #	External Co-POP Naming
AV40	VSS
AV42	VSS
AW7	VSS
AW9	VSS
AW11	VSS
AW15	VSS
AW25	VSS
AW27	VSS
AW37	VSS
AW39	VSS
AW41	VSS
AY8	VSS
AY10	VSS
AY26	VSS
AY28	VSS
AY32	VSS
AY34	VSS
BA7	VSS
BA15	VSS
BA19	VSS
BA21	VSS
BA23	VSS
BA25	VSS
BA27	VSS
BA33	VSS
BA39	VSS
BB18	VSS
BB20	VSS

Ball #	External Co-POP Naming
BB26	VSS
BB28	VSS
BB38	VSS
BB40	VSS
BC15	VSS
BC19	VSS
BC21	VSS
BC23	VSS
BC31	VSS
BC37	VSS
BC39	VSS
BD18	VSS
BD20	VSS
BD26	VSS
BD28	VSS
BD38	VSS
BE15	VSS
BE17	VSS
BE19	VSS
BE21	VSS
BE25	VSS
BE27	VSS
AR1	ZQ_1
BK40	ZQ_2
AH2	ZQ_3
BL41	ZQ_4

§

22 Package Information

The SoC comes in Flip-Chip Ball Grid Array (FCBGA) package and consists of a silicon die mounted face down on an organic substrate populated with solder balls on the bottom side. Care should be taken to avoid contact with the package inside this area.

Refer to *Cherry Trail - T SoC, Package Mechanical Drawing (PMD) and Ball Map – Ballout, Signal, and Mechanical Package* for details on package mechanical dimensions and tolerance, as well as other key package attributes.

22.1 SoC Attributes

Table 187. SoC Attributes

	Category	MSP-T4	VMS-T3	Co-POP	MSP-T4 Refresh
Package	Type	17x17mm Type 4	17x17mm Type 3	15x15mm Type4	17x17mm Type 4
	IO count	628	378	378 - Bottom 212 - Top	628
	Core Process (nm)	14	14	14	14
	Ball count	1380	592	1178 - Bottom 396 - Top	1380
	ball pitch	0.4mm	0.65mm	0.4mm	0.4mm
	Z-height	0.937mm	1.002mm	0.721mm[1]	0.937mm

Note: 1.This is PRE-SMT package height. The POST-SMT Z-height is TBD.

22.2 Package Diagrams

Figure 122.Package Mechanical Drawing for VMS-T3 (Part 1)

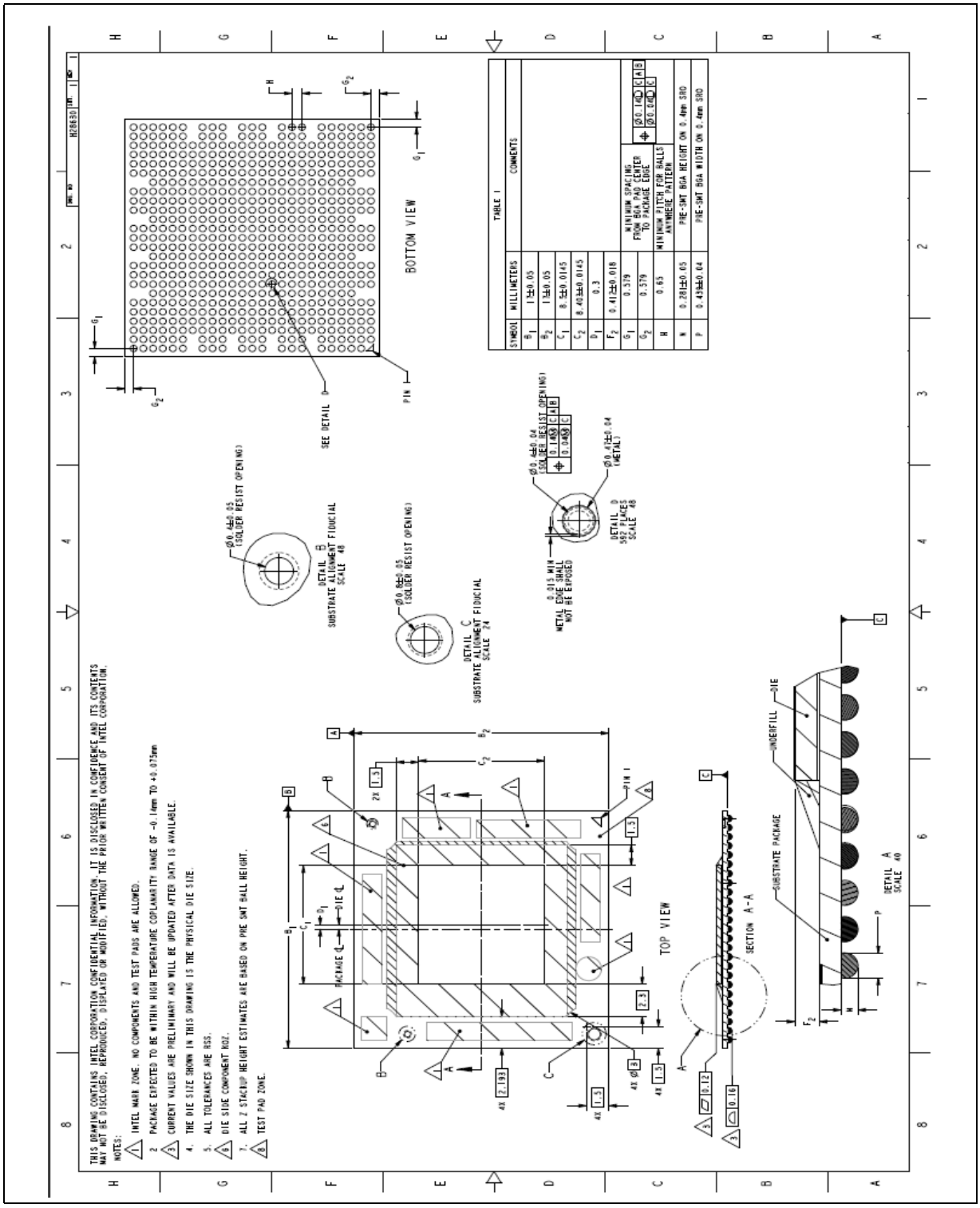


Figure 123.Package Mechanical Drawing for VMS-T3 (Part 2)

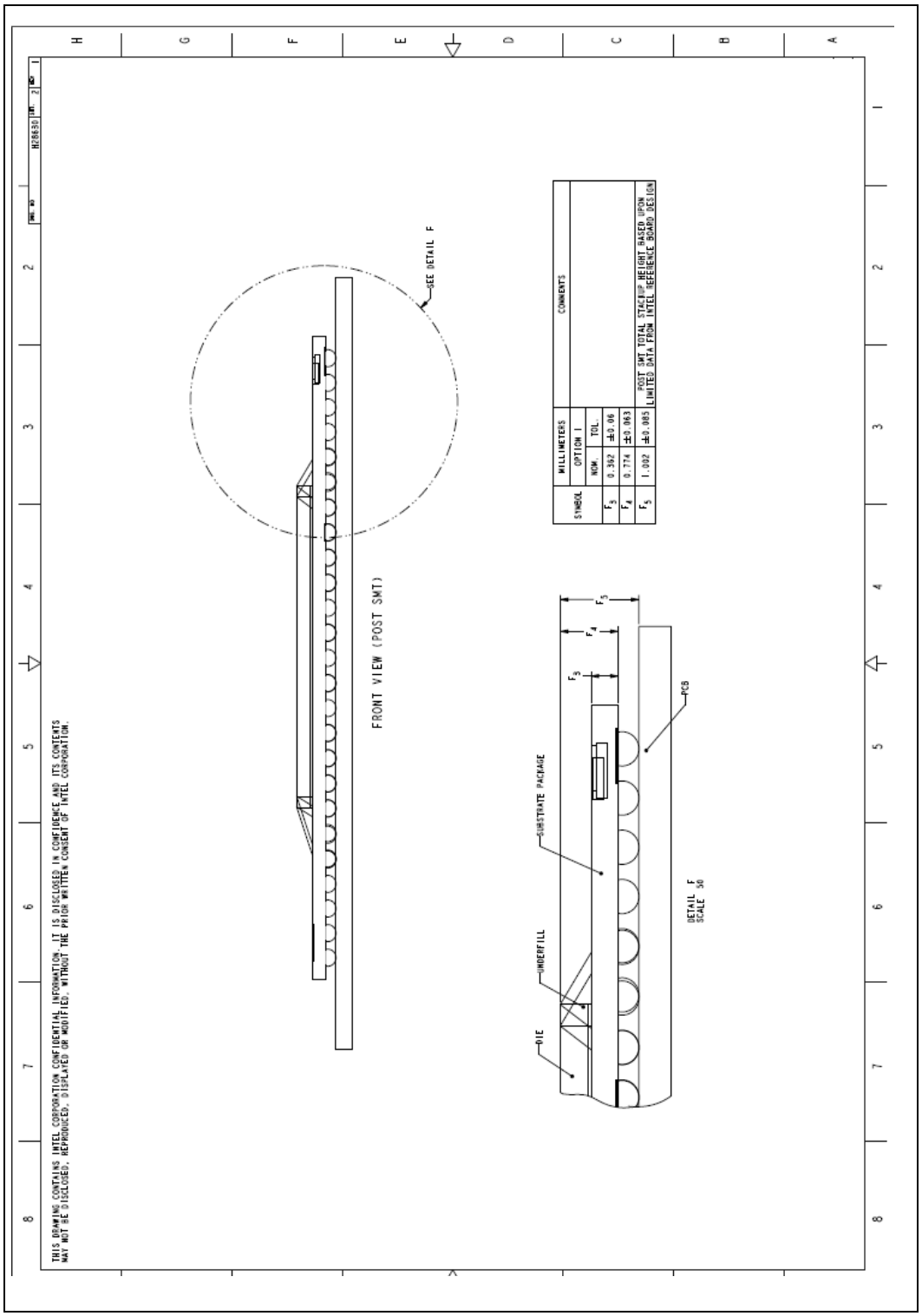


Figure 124.Package Mechanical Drawing for MSP-T4 (Part 1)

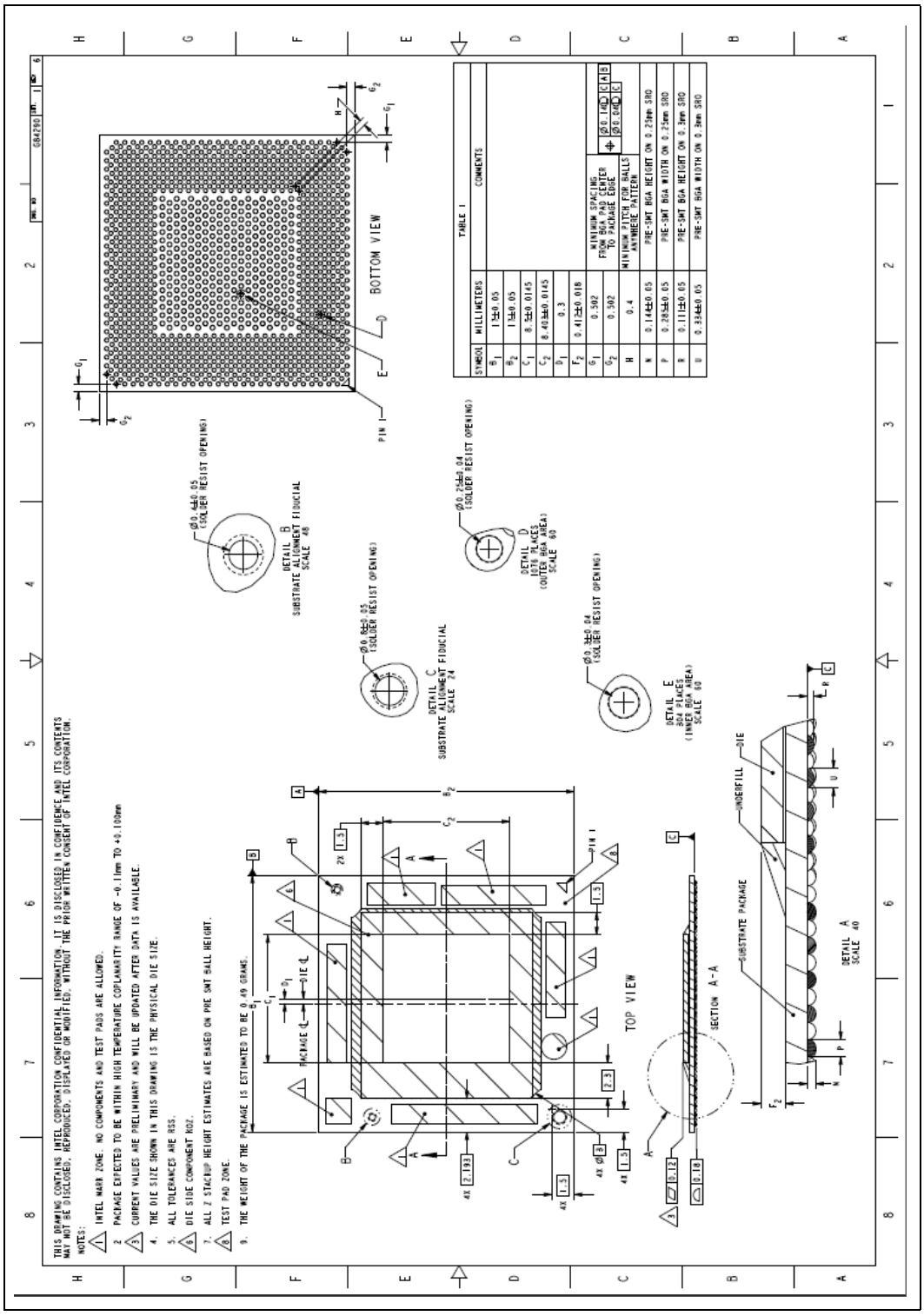


Figure 125.Package Mechanical Drawing for MSP-T4 (Part 2)

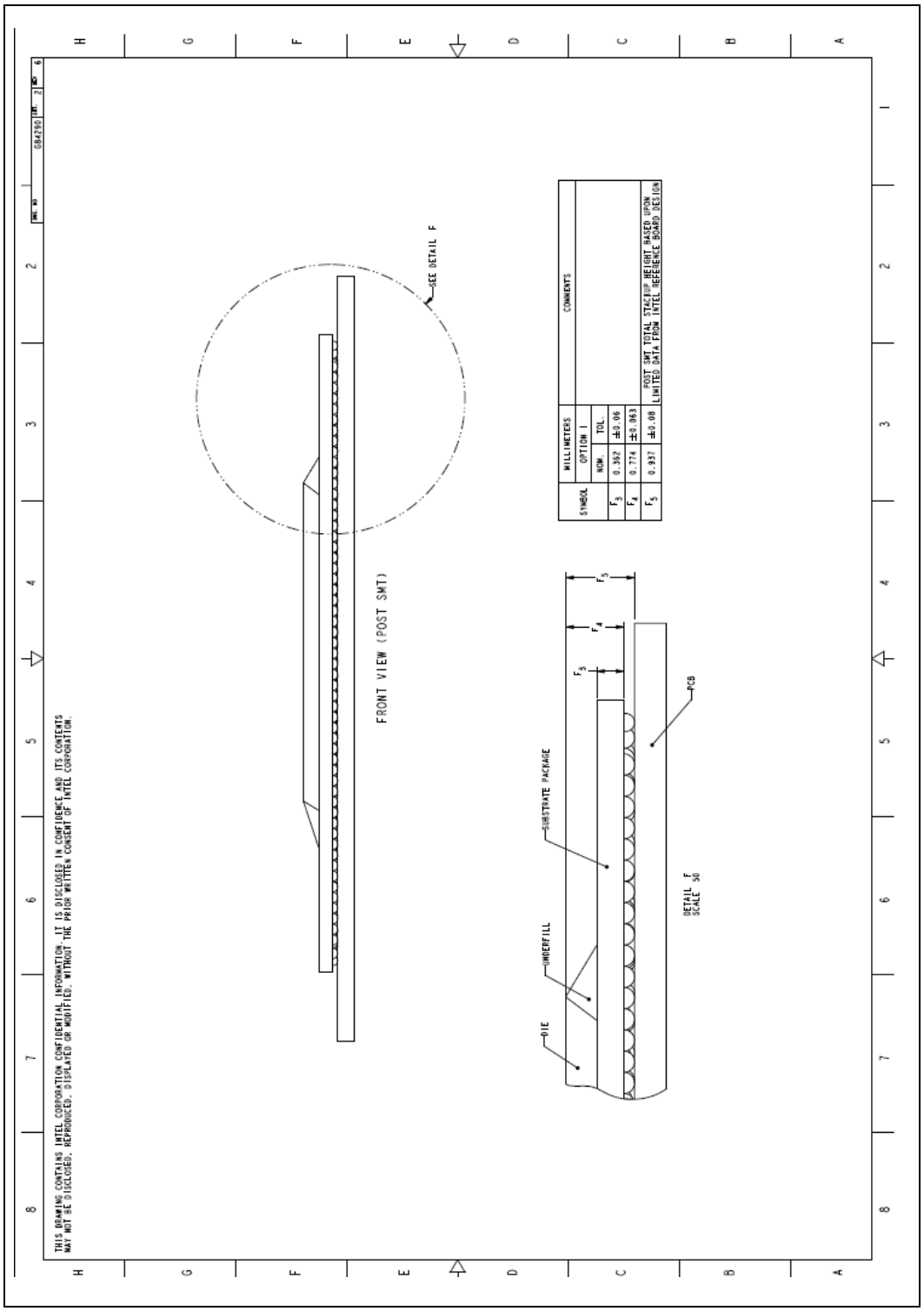


Figure 126.Package Mechanical Drawing for Co-POP (Part 1)

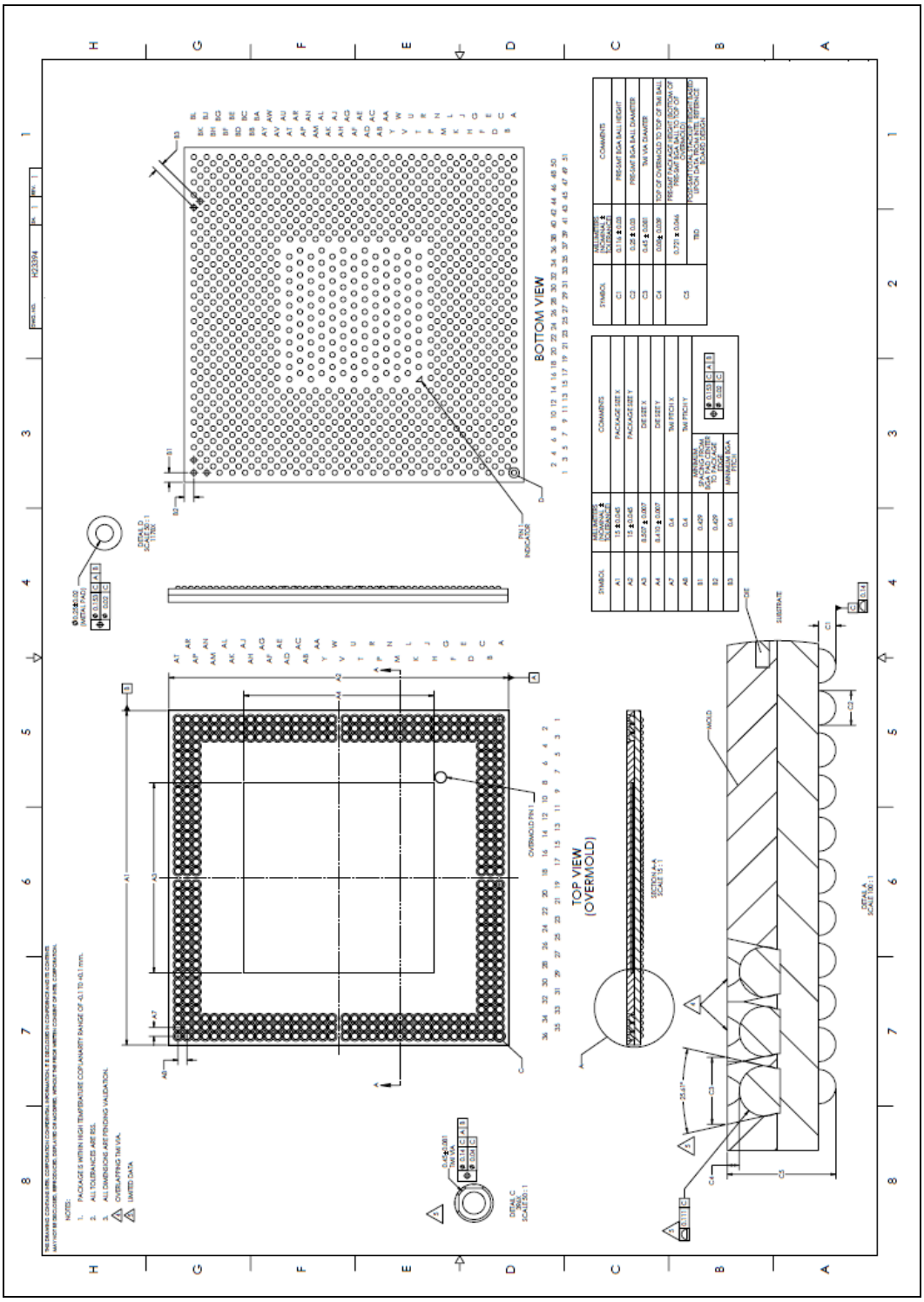
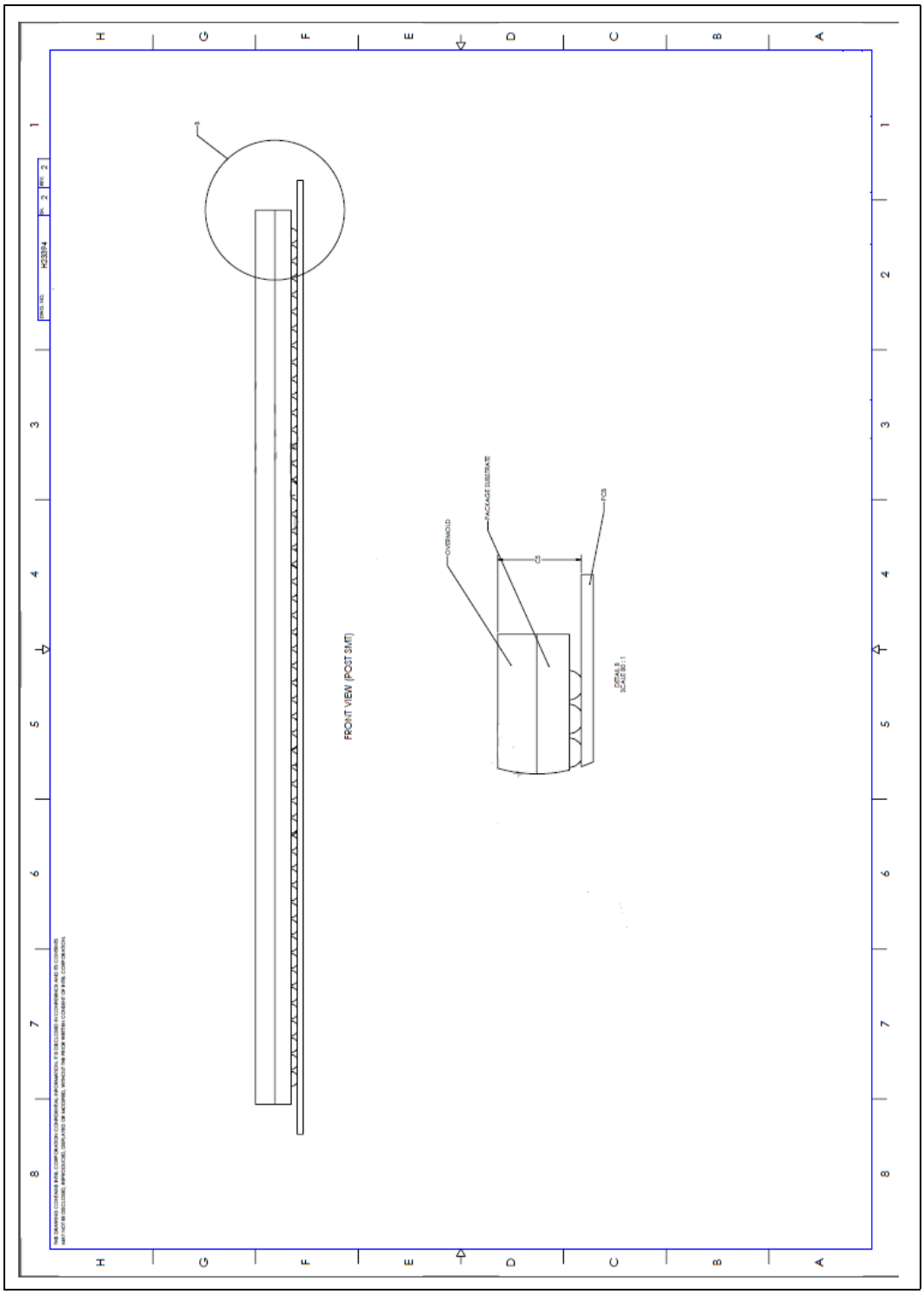
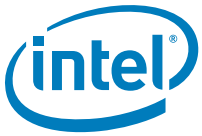


Figure 127.Package Mechanical Drawing for Co-POP (Part 2)





§