Delta Lake 1S Server Design Specification

1v00

Authors:

Michael Haken, Mechanical Engineer, Facebook

Thoon KY, Hardware Engineer, Facebook

Anthony Chan, Power Engineer, Facebook

Ben Wei, Software Engineer, Facebook

Yueming Li, Thermal Engineer, Facebook

Yao Rong, Hardware System Engineer, Facebook

Todd Westhauser, Hardware Engineer, Facebook

Kiran Vemuri, Hardware Engineer, Facebook

Pavan Shetty, Power Engineer, Facebook

1 Copyrights and Trademarks

Intel and Xeon are registered trademarks of the Intel Corporation in the U.S. and/or other countries.

Texas Instruments™ is a trademark of Texas Instruments Incorporated.

Tiva™ is a trademark of Texas Instruments Incorporated.

2 License

Your use of this Specification may be subject to other third party rights. THIS SPECIFICATION IS PROVIDED "AS IS." The contributors expressly disclaim any warranties (express, implied, or otherwise), including implied warranties of merchantability, non-infringement, fitness for a particular purpose, or title, related to the Specification. The Specification implementer and user assume the entire risk as to implementing or otherwise using the Specification. IN NO EVENT WILL ANY PARTY BE LIABLE TO ANY OTHER PARTY FOR LOST PROFITS OR ANY FORM OF INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION OR ITS

GOVERNING AGREEMENT, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND WHETHER OR NOT THE OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."

3 Scope

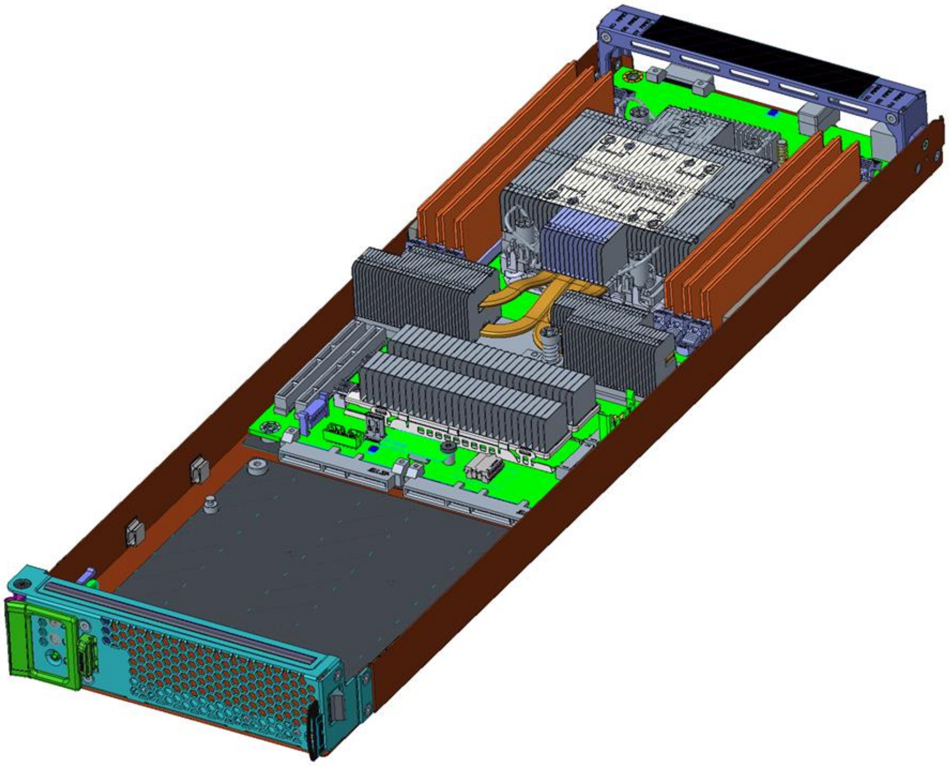
This specification describes the design of the Delta Lake 1S server based on 3rd Gen Intel Xeon Scalable processors (formerly codename Cooper Lake) Intel Xeon.

4 Contents

5 Overview

Note: All processor names mentioned in this doc refer to 3rd Gen Intel Xeon Scalable processors (formerly codenamed Cooper Lake processor and Cedar Island platform). Details on available Intel Xeon Scalable processors can be found at http://ark.intel.com/

This document describes a single socket server design based on Intel Xeon the 3rd Gen Intel Xeon Scalable processor, which is referred to hereinafter as Delta Lake 1S server.



The Delta Lake 1S server is designed to use Intel Xeon 3rd Gen Intel Xeon Scalable processors utilizing the performance and advanced intelligence of Intel Xeon Scalable processors. It works together with Intel's C620 Chipset series Platform Controller Hub.

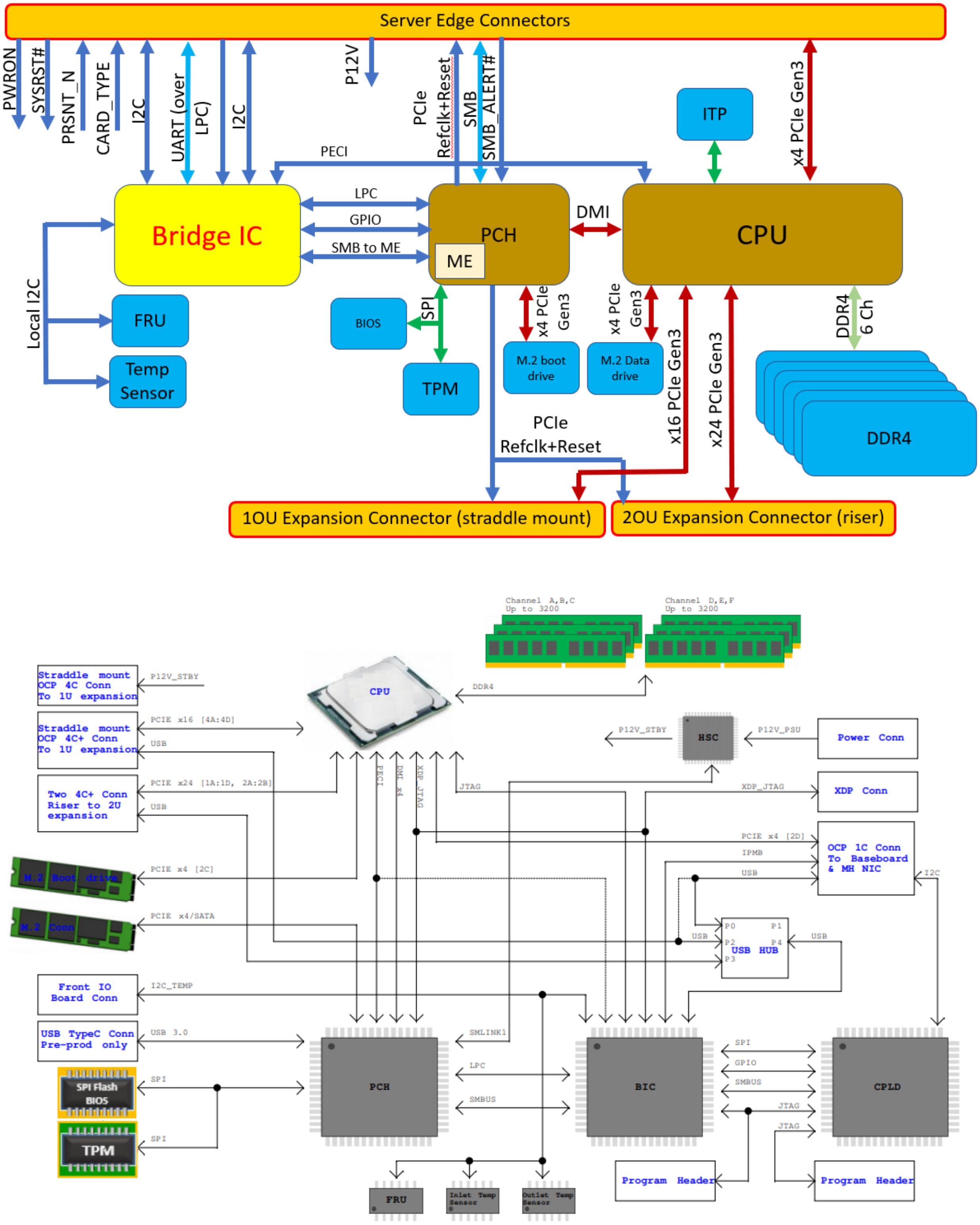


Figure 5-1: Delta Lake 1S Server Block Diagram

The Delta Lake 1S server has a cable to connect both high speed and low speed signals from the 1C connector on Delta Lake to the Integrated Baseboard which include a x4 PCIe channel as well as sideband signals to connect to a Multi Host (MH) NIC that also connects to a Multi Host (MH)

NIC. Delta Lake also has a power connector to connect to a vertical power distribution board (PDB). The main interfaces to the server board through the Server Edge Connectors are as follows (from figure 5-1):

- PCIe Gen3 connections to the Network Interface Controller (NIC) (4x lanes)
- A Universal Asynchronous Receiver/Transmitter (UART)
- 1x 1MHz IPMB for communication between BMC (outside of Delta Lake) to BIC (Bridge IC on Delta Lake)
- 1x 400KHz I2C for BMC (outside of Delta Lake) to control devices on Delta Lake.
- A high-speed USB interface between BMC to BIC (on Delta Lake). Delta lake has a hub to enable the BMC USB to communicate with BIC on expansion boards.
- Status and control signals for power button, power enable and power good.
- A serial bit stream GPIO.

The server base option will have 2 M.2 SSD drives. One is used as boot drive and the other is used as data drive. The x4 PCIe link to the boot drive shall be connected to the PCH, and the data drive shall be connected to the CPU. The data drive has a x4 PCIe link directly to the CPU. The recommended minimum capacity of the boot drive is 256GB. The system can support M.2 drives of 22110 or 2280 form factor.

The Delta Lake 1S server shall connect to an external NIC on the platform through its PCIe. The server will utilize the remaining PCIe lanes to enable high bandwidth connections to 1U and 2U expansion boards. The expansion boards allow several configurations in Data Center with the server module detailed in section 5.1.1.

Currently we are provisioning 2 configs for PCIe connectivity.

Table 5-1: PCIe connectivity configurations

PCIe lanes with CPX-6 CPU as root port		
	Class 1 (primary configuration)	Class 2 (special configuration)
Multi-host shared NIC on Baseboard	x4 (out of x16 as form factor max)	x4 -unused
Riser	x24 (out of x32 as form factor max)	x24 (out of x32 as form factor max)
Onboard M.2	x4	x4
Front expansion	x16 (out of x24 as form factor max)	X16 to dedicated NIC
PCIe lanes with PCH as root port		
Onboard M.2 boot drive	X4	X4

The Delta Lake 1S server has a dedicated hot swap controller (HSC) to enable the insertion/removal of the server blade into/from a live power bus while providing important telemetry such as voltage, current, and power (VIP). The telemetry is used by the platform for power management purposes such as imposing power limits and triggering of throttle events to force the CPU to the lowest power state.

The Delta Lake 1S server also supports an Advanced Configuration Power Interface (ACPI)-compliant power button and reset signals from the platform.

A Bridge IC(BIC) is used as the management controller on the 1S server and the bridge between the BMC on base board, to devices on Delta Lake 1S CPU card. Examples of devices are CPU, and PCH. The Bridge IC manages the 1S server on behalf of the BMC on the platform and bridges the BMC and Intel PCH internal management controller Intel Manageability Engine. To maximize the communication bandwidth between the BMC and the Bridge IC, a dedicated point-to-point I^2C bus shall be used. IPMB is running on this I2C bus to allow both devices on the bus to initiate communication.

Delta Lake 1S server's Field Replaceable Unit (FRU) EEPROM and thermal sensors are connected to the Bridge IC's other I^2C buses. There are multiple General Purpose Input/Output connections (GPIOs) between the Bridge IC and Intel Xeon 3rd Gen Intel Xeon Scalable processor for error reporting and other management purposes. The Intel PCH chipset and Intel Manageability Engine is connected to the Bridge IC via an I^2C interface so that the Bridge IC can poll information from the Intel Manageability Engine. A Low Pin Count (LPC) bus between the Bridge IC and the Intel PCH is connected to enable in-band communications. The BMC can access the Delta Lake 1S server's thermal sensors, FRU, Intel PCH's GPIOs and Intel Manageability Engine via the Bridge IC with standard IPMI commands.

BIOS, network controller and boot ROM of Bridge IC on the Delta Lake 1S server, can be updated from in-band connectivity by the PCH or out-of-band thru BMC. While other firmware (CPLD and VR chips) are programmable from out-of-band connectivity by the Bridge IC and BMC.

All configurations of Delta Lake will support Intel's Platform Firmware Resiliency as the Platform Root of Trust solution. The Altera MAX10 FPGA will enforce secure boot and secure firmware updates for the BIOS and CPLD firmware on Delta Lake. This is explained in more detail in Section 11.22.

5.1 Use cases

This section describes major use cases of Delta Lake 1S Server card in the context of Yosemite V3 platform. This is to provide context for proper implementation of Delta Lake 1S server card.

The Yosemite V3 platform detail is covered in Yosemite V3 platform Specification.

The 1U and 2U Expansion Card detail is covered in Expansion cards specification.

5.1.1 1U/2U Multi-Host Use cases (Class 1 use case)

In this class of use cases, multiple Delta Lake 1S server cards share common baseboard BMC, and a multi-host NIC through baseboard. The Delta Lake 1S server can be in 1U or 2U for different amount of front expansion and host to NIC ratio. The front expansion is optional in 1U use case.

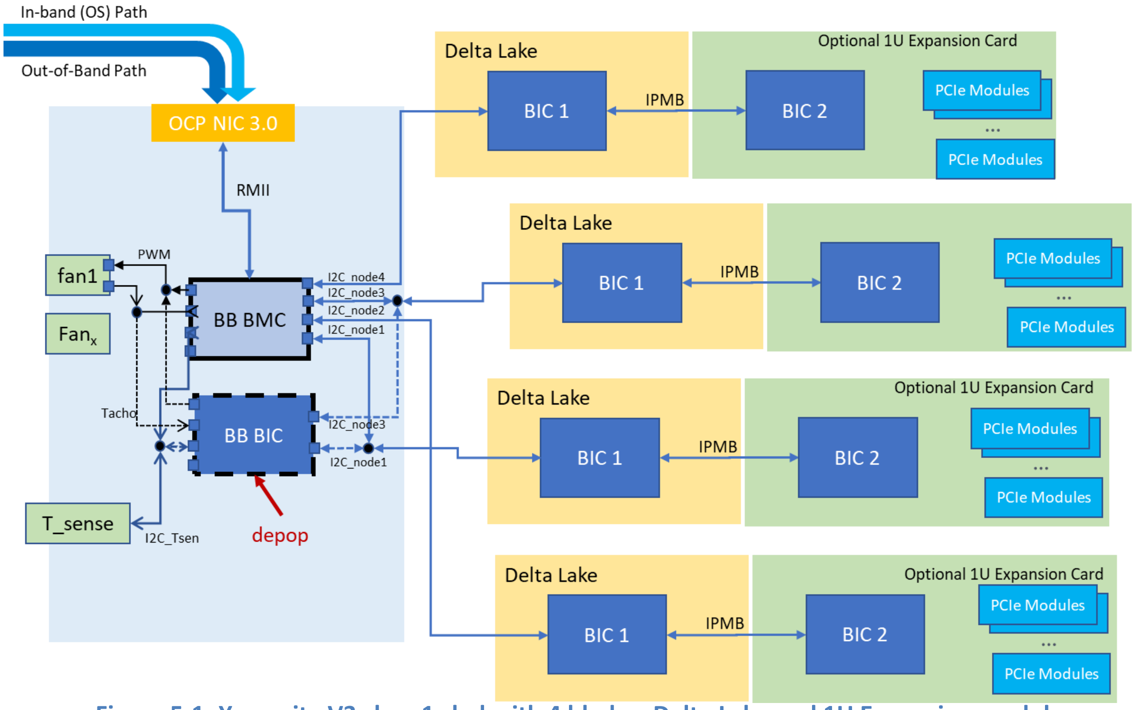


Figure 5-1: Yosemite V3 class 1 sled with 4 blades: Delta Lake and 1U Expansion module

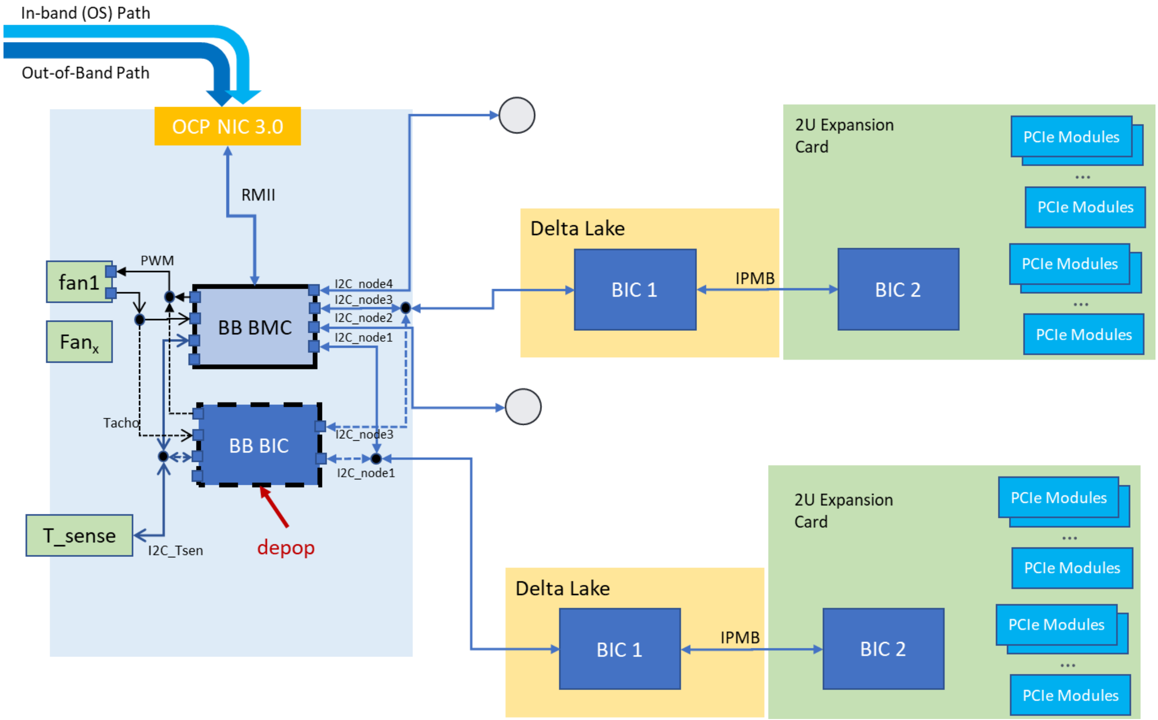


Figure 5-2: Yosemite V3 class 1 sled with 2 blades: Delta Lake and 2U Expansion module

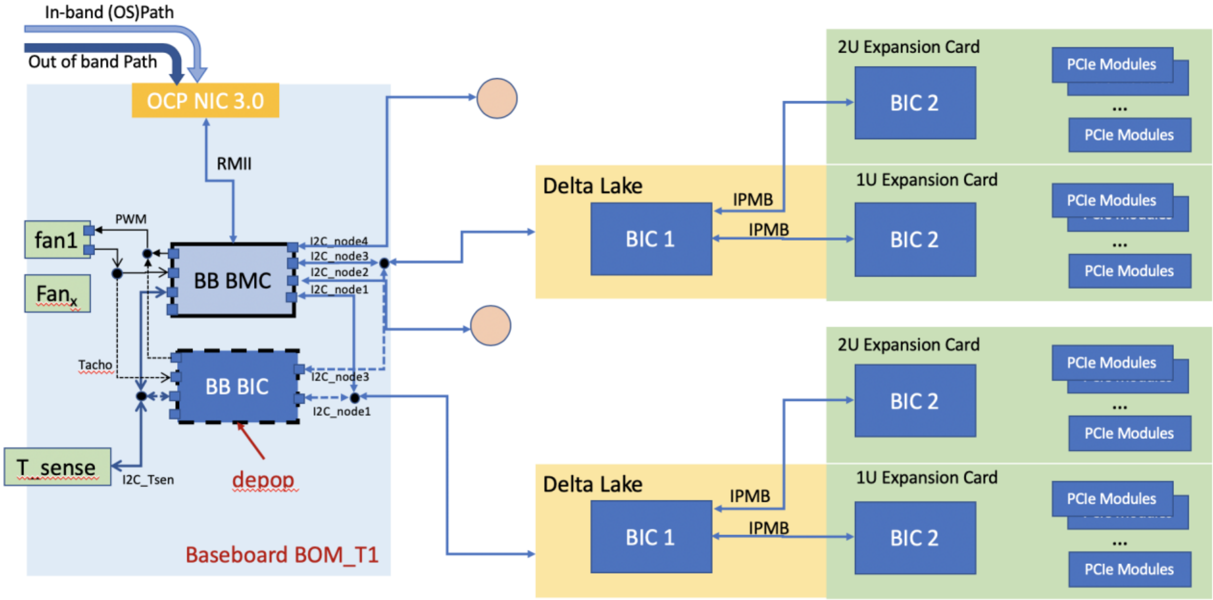


Figure 5-3: Yosemite V3 class 1 sled with 2 blades: Delta Lake, 1U and 2U Expansion modules

5.1.2 2U Single Host 1S Server Card (Class 2 use case)

In this class of use cases, each Delta Lake has its own BMC and single host NIC connection (100G is shown as an example). BMC for each Delta Lake is on Network Expansion Card connected to Delta Lake.

2x Delta Lake server cards share same baseboard for chassis management purpose. There is a BIC on baseboard for chassis management purpose. There is no NIC, and no BMC on baseboard in this use case.

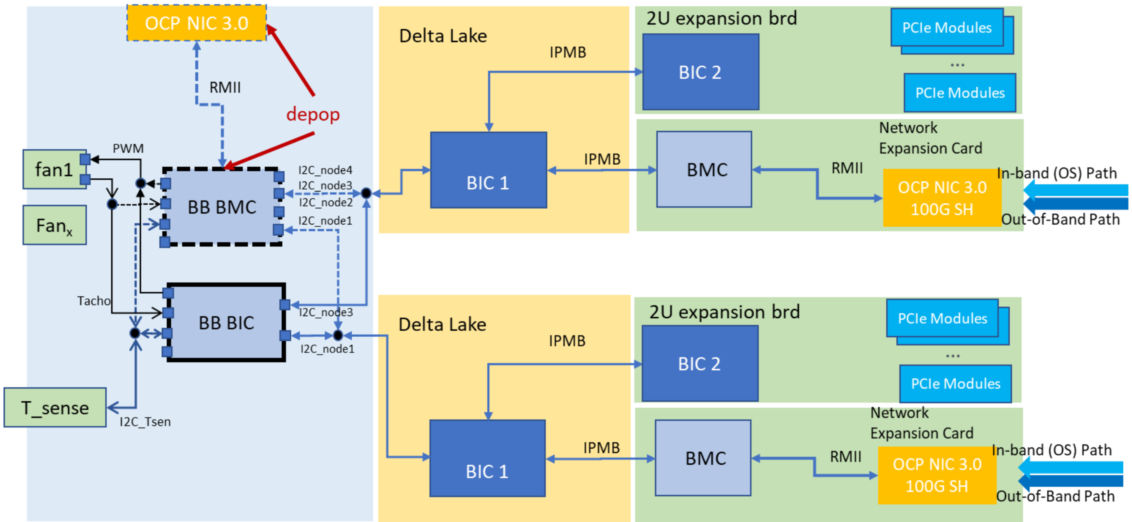


Figure 5-4: Yosemite V3 class 2 sled with 2 blades: Delta Lake, 2U Expansion module and single host NIC

6 Functional

6.1 CPU

Delta Lake 1S server will be built with an Intel Xeon Processor CPX-63rd Gen Intel Xeon Scalable processors. This processor is a multicore LGA part that includes memory controllers and high-speed IO interfaces. The CPU is socketed and is a Field Replaceable Unit (FRU) along with its heatsink. The CPX-6 CPU is available in various power profiles. The exact CPU power SKU along with the needed power supply configuration will be decided after testing.

6.2 PCH

The Intel C620 Chipset series Platform Controller Hub (PCH) on Delta Lake works in tandem with the CPU. It generally provides extensive IO support for the CPU that includes

- ACPI power management Logic Support
- Integrated Serial ATA host controller
- xHCI USB controller with SuperSpeed USB 3.0 ports
- Direct Media Interface
- PCIe for peripheral and storage devices
- SPI and eSPI interface
- Flexible I/O
- GPIO
- LPC and SMBus

And many more. Refer to respective EDS from Intel with regards to the PCH.

6.3 Memory

The Delta Lake 1S server uses DDR4 memory bus, expected to run at 2933MT/s minimum, with design target to hit maximum speed of 3200MT/s.

Delta Lake 1S server shall support 1x DDR4 R-DIMM per channel on 6 channels.

Design can support all Intel 3DX-Point based NVDIMM's configuration through BOM configuration.

Table 6-1: Memory Matrix

Config	Channels Used	DIMMs	Rank	DRAM	Note
64GB	4	16GB	2R, x8 DRAM	8Gb	
96GB	6	16GB	2R, x8 DRAM	8Gb	
192GB	6	32GB	2R, x8 DRAM	16Gb	For testing worst case configuration for thermal design

Memory configs expected on Delta Lake (all going up to 3200MT/s, 1DIMM per channel)

6.4 Debug headers

The Delta Lake 1S server shall support traditional ITP/XDP debug headers.

6.5 Boot drive

The baseline version of Delta Lake 1S server has 1 M.2 solid-state boot drive in 2280 or 22110 form factors. This drive is connected to CPU through PCIe. A M.2 NVMe SSD is required as a boot device and for logging.

6.6 Expansion systems

The Delta Lake 1S server module allow certain level of expansion through additional PCIe lanes that are available on the CPU in form of a riser card and a front-end straddle mount card. Due to a wide possibility of expansion system that can be made available, the specification here is only to indicate what level of connectivity is made to these expansion systems on these systems.

Riser card: (pin definition similar to OCP NIC 3.0, but not all pins are adopted)

- 24 lanes of PCIe, 2 PCIe ref clocks
- Reset (from PCH)
- I2C (from BIC), Interrupts
- USB (from BMC), UART (from Delta Lake CPLD)
- PWRBRK
- 12V and 3.3V_STBY
- RISER CARD TYPE DETECTION (3x)

- Power Enable/Power Good

Front-end straddle mount: (to be pin-out like LFF OCP NIC 3.0 with all pins adopted)

- 16 lanes PCIe, 1 PCIe ref clocks
- Reset (from PCH)
- I2C (from BIC), Interrupts
- USB (from BMC), UART(from Delta Lake CPLD)
- Wake, PWRBRK
- 12V and 3.3V_STBY
- Expansion PRSNT
- Expansion SLOT ID

Take note that the intent for these expansion systems is expected to be NOT hot-swappable. In addition, designer for these expansion systems need to ensure the power and thermal system of the entire module is maintained. Refer to Delta Lake 1S expansion system for more details.

6.7 FRU EEPROM

The Delta Lake 1S server includes a 128Kbits I^2C-accessible Electrically Erasable Programmable Read-Only Memory (EEPROM). The EEPROM is accessible from the platform via the Bridge IC. The EEPROM contains the Field Replaceable Unit Identification (FRU ID) information and any additional configuration information that may be required. The FRU ID is formatted in accordance with the IPMI Platform Management FRU Information Storage Definition document.

The EEPROM must contain the following entries:

- Board Manufacturer
- Board Name
- Board Serial Number
- Board Part Number
- Product Manufacturer
- Product Name
- Product Part Number
- Product Serial Number
- Product Asset Tag
- Product Build: e.g. EVT, DVT, PVT, MP
- Product Version: e.g. C1
- Manufacturing Date and Time
- Manufacturing Lot Code: (preferred, but optional)
- Manufacturing Work Order: (preferred, but optional)
- PCB Revision
- SoC Model Name/Number
- SoC Revision
- SoC Tj_{MAX} (Maximum Junction Temperature)

7 Mechanical

7.1 Mechanical Outline

The mechanical CPU blade module comes in 1U and 2U form factors as seen in Figure 6-1 and 6-2, respectively.

The Delta Lake 1S Server card interfaces to the platform using 1x SFF-TA1002 1C for signal and FCI's ortho Power Blade Ultra connector (TE's MBXLE ortho) for power.

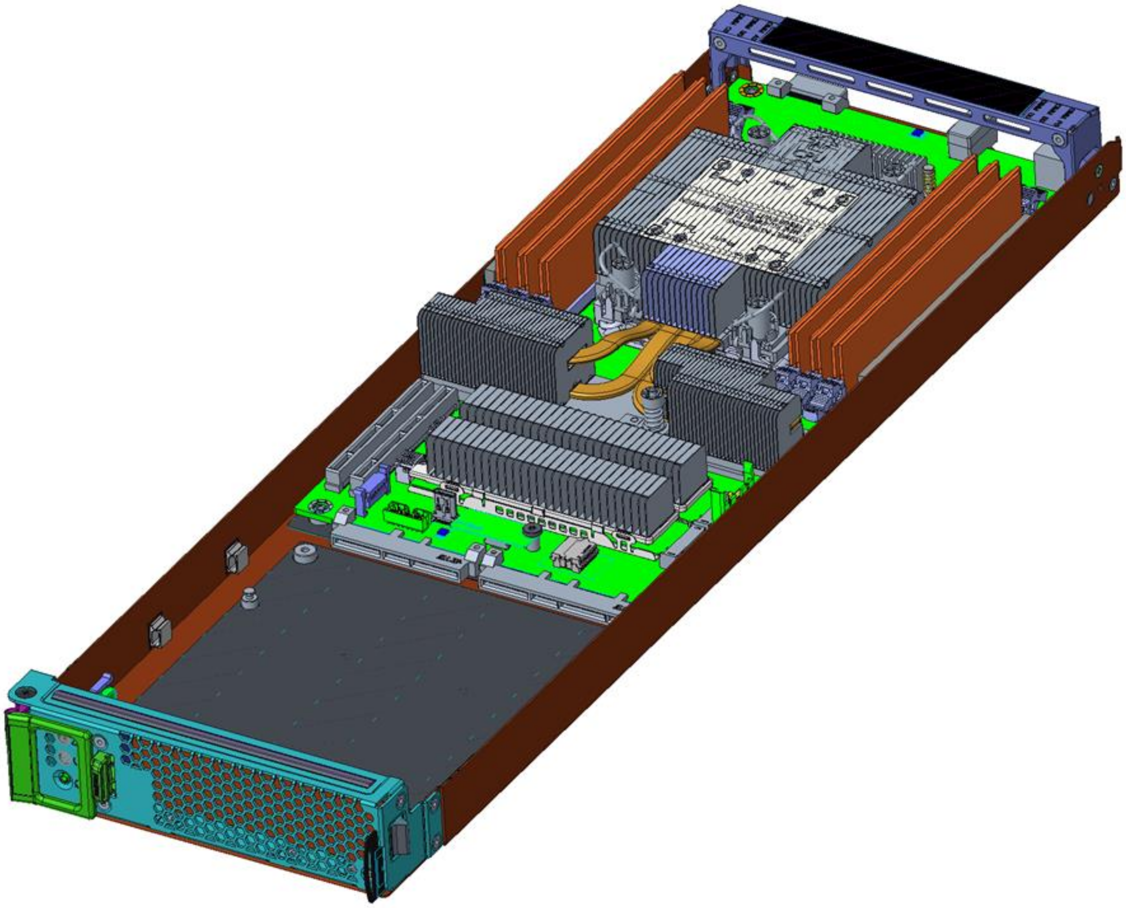


Figure 6-1: Yosemite V3, 1U Delta Lake, ISO view

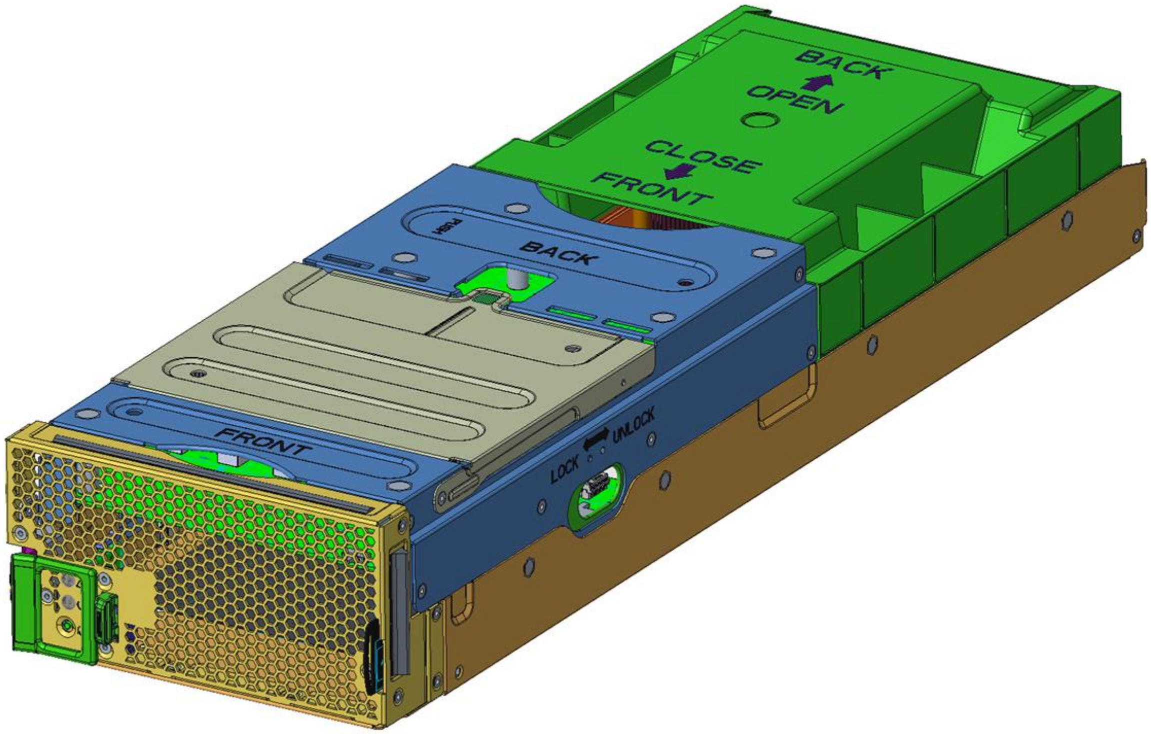


Figure 6-2: Yosemite V3, 2U Delta Lake, ISO view

The overall dimentions of the server card is 321.4mm x 157.0mm. The dimensions of the blade module is 533.5mm x 164.0mm, as seen in Figure 6-3. The 1U blade height is 40.8mm while the 2U blade height is 82.2mm, as seen in Figure 6-4.

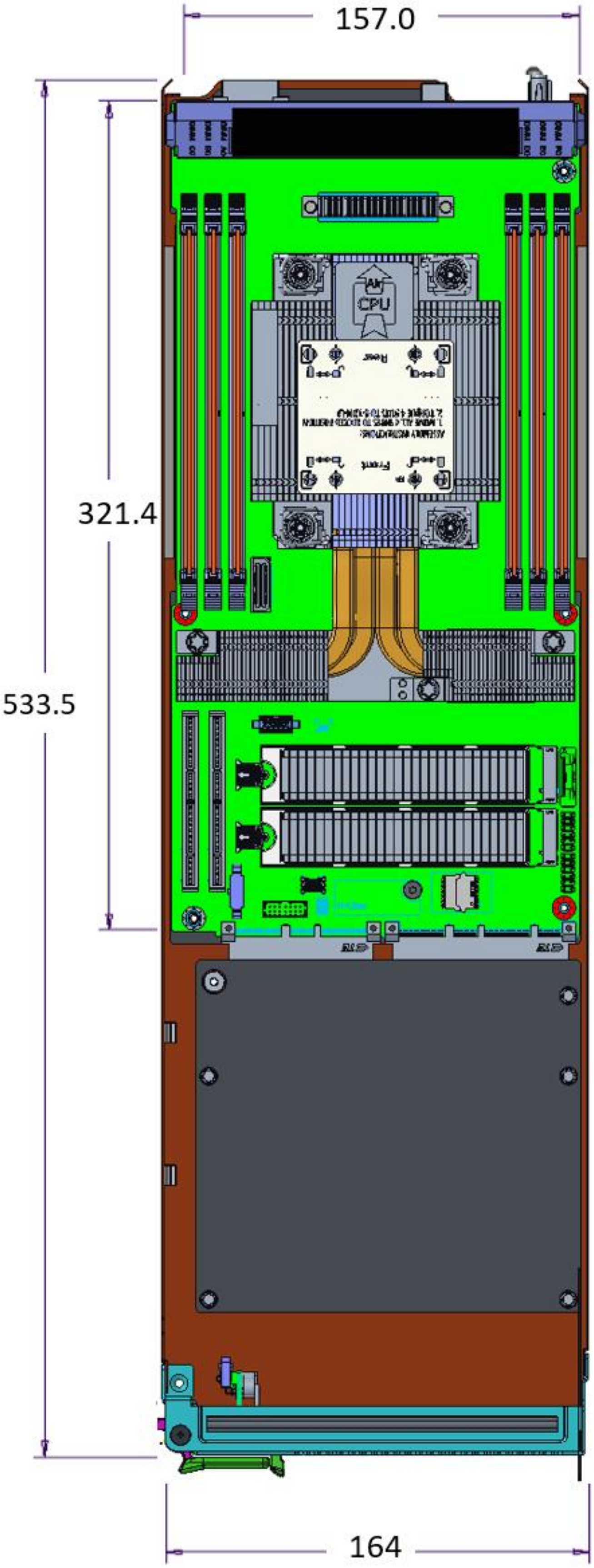

157.0
321.4
533.5
164
Air
CPU
Rear
Front

Figure 6-3: Yosemite V3, Delta Lake, Mechanical Outline

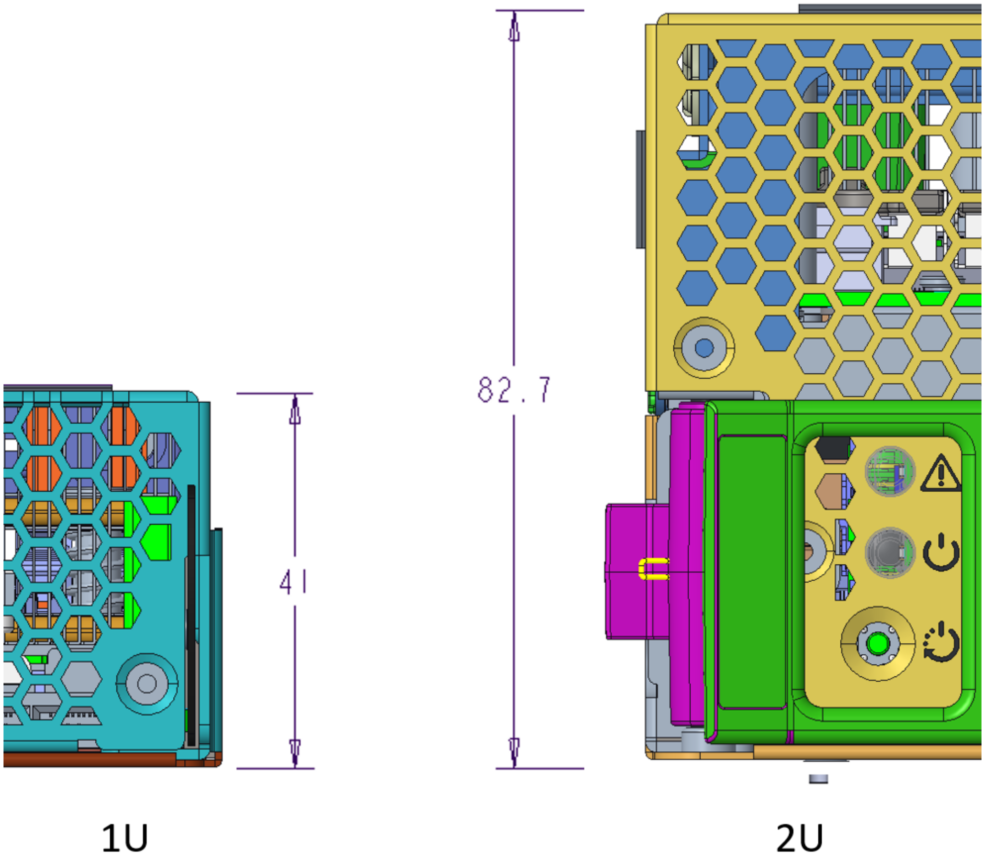


Figure 6-4: Yosemite V3, Delta Lake, Mechanical Outline

The front of the Delta Lake blade receives the expansion module. There are several possible expansion card options for 1U and 2U blade form factors. The expansion card is defined in the OCP Yosemite V3 Expansion Card Specification. An example of an expansion card can be seen in Figure 6-5.

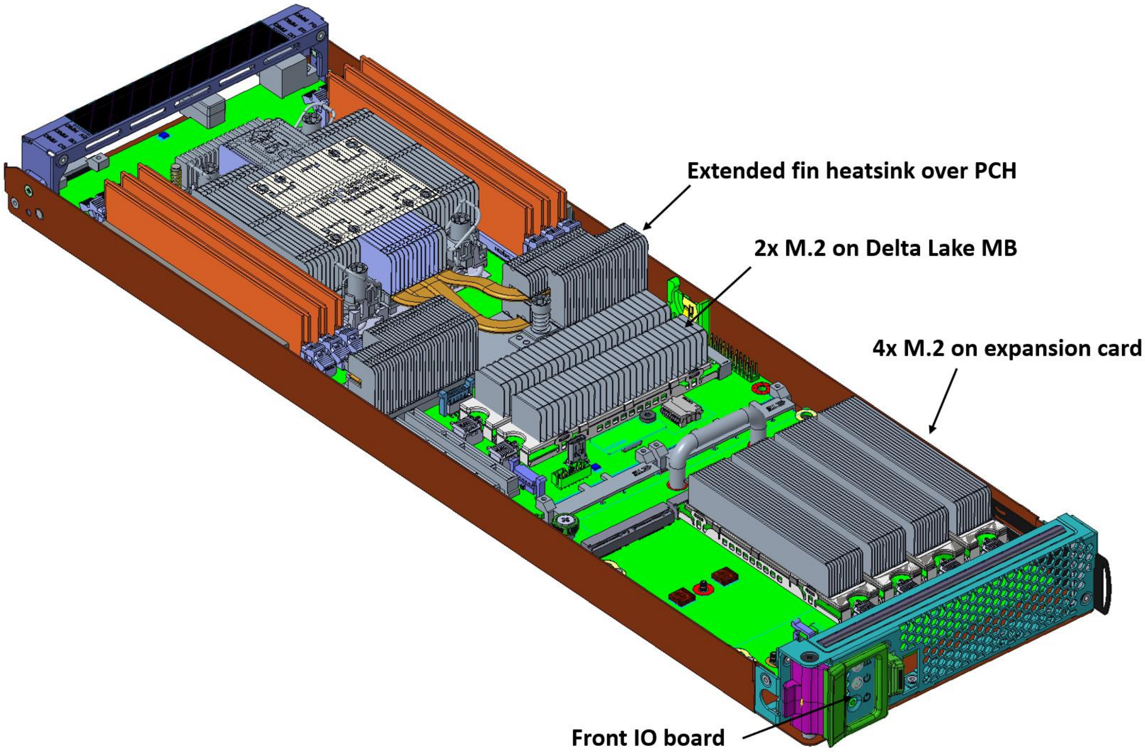


Figure 6-5: Yosemite V3, Delta Lake with 1OU Expansion Card for Reference

7.2 Platform Design

Platform design details are not discussed in this specification. Refer to the OCP Yosemite V3 Platform Design Specification for more detail.

8 Thermal

To meet thermal reliability requirement, the thermal and cooling solution should dissipate heat from the components when system operating at its maximum TDP (Thermal Design Power). The thermal solution should be found by setting a high-power target for initial design in order to avoid redesign of cooling solution; however, the final thermal solution of the system should be most optimized and energy efficient under data center environmental conditions with the lowest capital and operating costs. Thermal solution should not allow any overheating issue for any components in system. CPU or memory should not throttle due to any thermal issue under following environment.

- Inlet temperature lower than or equal to 35°C, and 0-inch H2O datacenter pressure with all FANs in each thermal zone running properly
- Inlet temperature lower than or equal to 35°C, and 0.001-inch H2O datacenter pressure with one FAN (or one rotor) in each thermal zone failed

8.1 Data Center Environmental Conditions

This section outlines Facebook data center operational conditions.

8.1.1 Location of Data Center/Altitude

Maximum altitude is 6,000 ft above sea level. Any variation of air properties or environmental difference due to the high altitude needs to be deliberated into the thermal design.

8.1.2 Cold-Aisle Temperature

Data centers generally maintain cold aisle temperatures between 18°C and 30°C (65°F to 85°F). The mean temperature in the cold aisle is 24°C with 3°C standard deviation. The cold aisle temperature in a data center may fluctuate minutely depending to the outside air temperature. Every component must be cooled and must maintain a temperature below its maximum specification temperature in the cold aisle.

8.1.3 Cold-Aisle Pressurization

Data centers generally maintain cold aisle pressure between 0 inches H_2O and 0.005 inches H_2O. The thermal solution of the system should consider the worst operational pressurization possible, which generally is 0 inches H_2O and 0.001 inches H_2O with a single fan (or rotor) failure.

8.1.4 Relative Humidity

Data centers usually maintains a relative humidity between 20% and 90%. The thermal solution must sustain uninterrupted operation of the system across the aforementioned RH range.

8.2 Server Operational Conditions

8.2.1 System Volumetric Flow

The unit of airflow (or volumetric flow) used for this spec is cubic feet per minute (CFM). The CFM can be used to determine the thermal expenditure or to calculate the approximate Delta T of the system. The thermal expenditure is quantified by the metric CFM/W, which is calculated by the following formula:

$$\text{Thermal Expenditure} = \frac{\text{System airflow}}{\text{Total system power consumption, including fans}} \quad [\text{CFM/W}]$$

The required airflow is 0.115 airflow per watt in the system level at sea level. The desired airflow per watt is 0.1 or lower in the system up to 35°C (95°F) ambient temperature at sea level for both 1U and 2U blade form factors. The cooling solution shall take the pre-heating effect from the expansion card into consideration if there is any.

8.2.2 Thermal Margin

The thermal margin is the difference between the maximum theoretical safe temperature and the actual temperature. The server blade design operates at an inlet temperature of 35°C (95°F) outside of the system with a minimum 5% thermal margin for every component on the card.

8.2.3 Upper Critical Threshold

The upper critical threshold (UCT) setting should allow the detection of abnormal thermal behaviors in the system. The UCT values for the sensors that are not used in Fan Speed Control (FSC) should use a 15% thermal margin from the worst experiment data. The UCT values for the sensors used in FSC, except for CPU, inlet, and outlet sensors, should use a 20% thermal margin from the worst experiment data.

8.2.4 Thermal Testing

Thermal testing must be performed at a low inlet temperature 15°C (59°F) and up to 35°C (95°F) inlet temperature to guarantee the design is free of thermal defect and has high temperature reliability. Rack-level airflow testing is recommended to evaluate rack level containment and ensure the thermal design will meet rack-level deltaT requirement.

8.3 CPU cooling solution

8.3.1 Passive Cooling Solution

The heat sink must be a thermally optimized design at the lowest cost. Passive cooling is desired. Heat sink installation must be uncomplicated. The heat sink fins should be aligned with the airflow direction. Heat sinks must not block debug headers or connectors and may require cutouts in order to avoid tall electrical components on the server card.

In 1U Delta Lake configuration, the heatsink must be within the footprint and include side walls on heatsink base to avoid delamination between heatsink fin and base as shown in Figure 8-1.

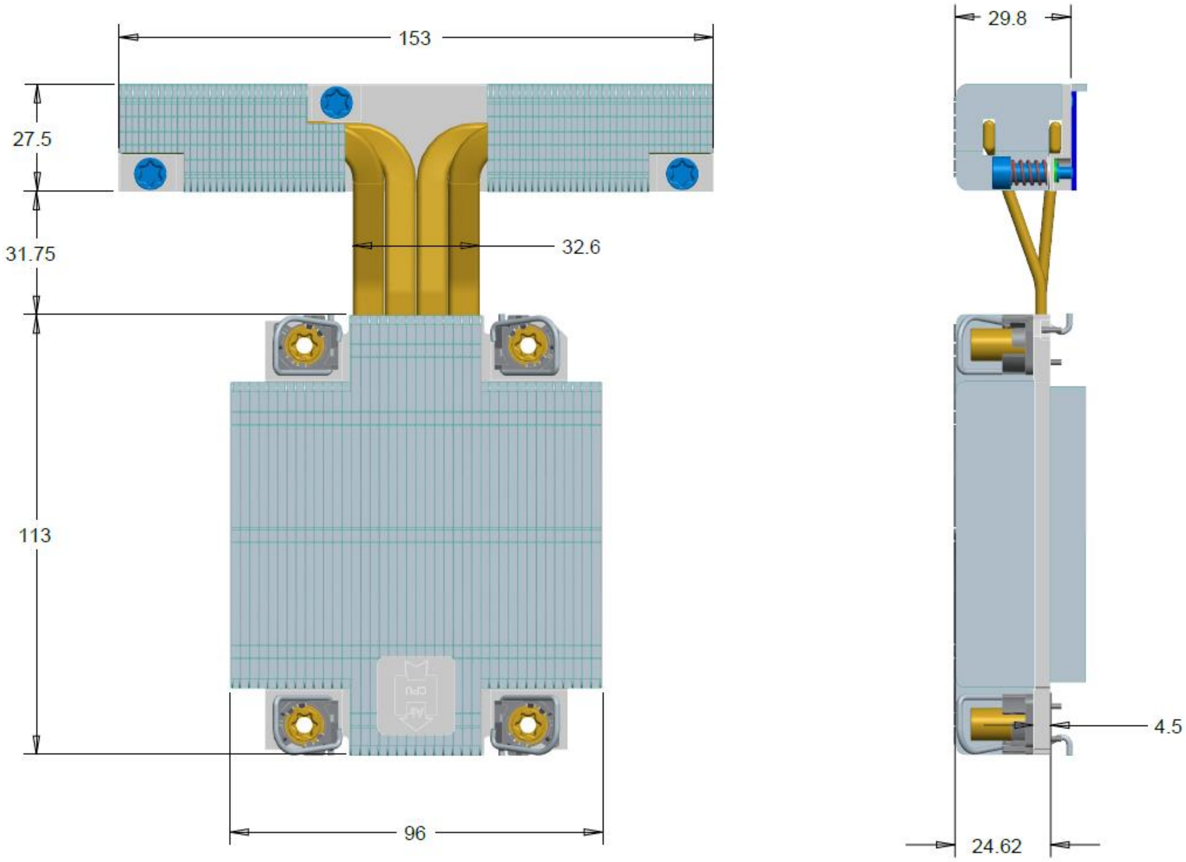


Figure 8-1: 1U Delta Lake Maximum Heatsink Footprint

In 2U Delta Lake configuration, the heatsink must be within the footprint as shown in Figure 8-2.

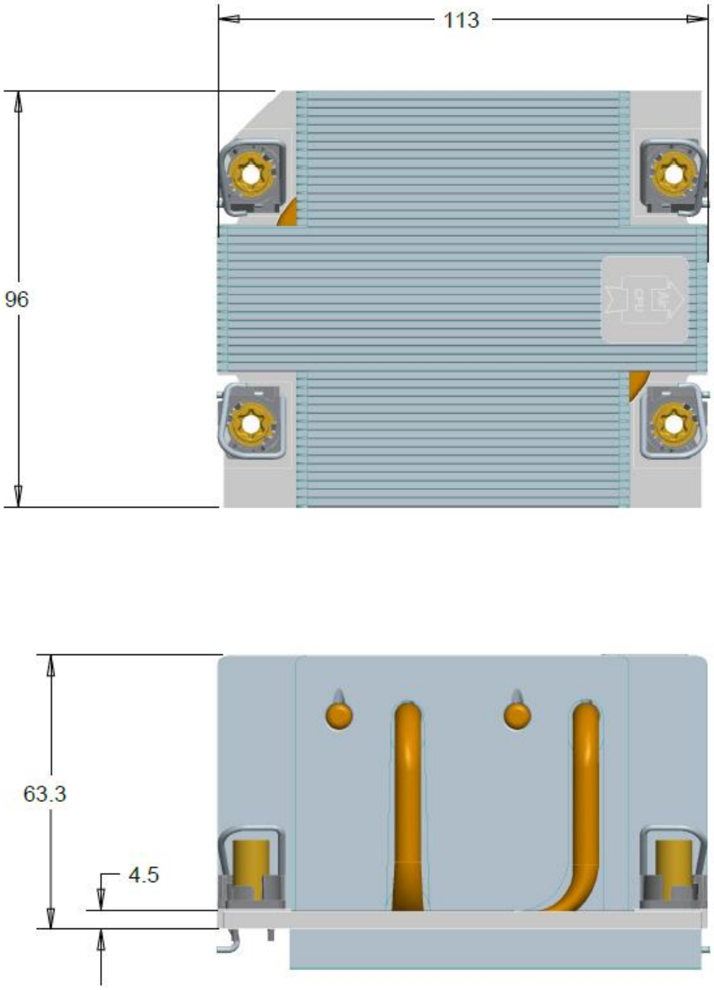


Figure 8-2: 2U Delta Lake Maximum Heatsink Footprint

8.4 Reliability Requirements

The operating life of the server card is a minimum of 3 years with target life of 5 years. All pertinent reliability data for CPU cooling solution and PCBA must be provided to meet the MTBF of 1.5 million hours at 30C and at a 90% confidence level.

8.5 M.2 Cooling Solution

Each card must support up to 2 M.2s. One boot drive and one high speed data drive in front of the CPU.

The cooling solution for M.2 cooling must be thermally optimized design at the lowest cost. Passive cooling is required with TIM and heat sink solution being desired. Integrated heatsink is desired to enable easy serviceability. The solution must maintain all M.2 component temperatures within their operational limits during all stress conditions. 6.7mm M.2 connectors must be used to allow spacing between integrated heatsink base and server board PCB.

Components under the boot M.2 must follow the keep out requirement as shown in Figure 8-3.

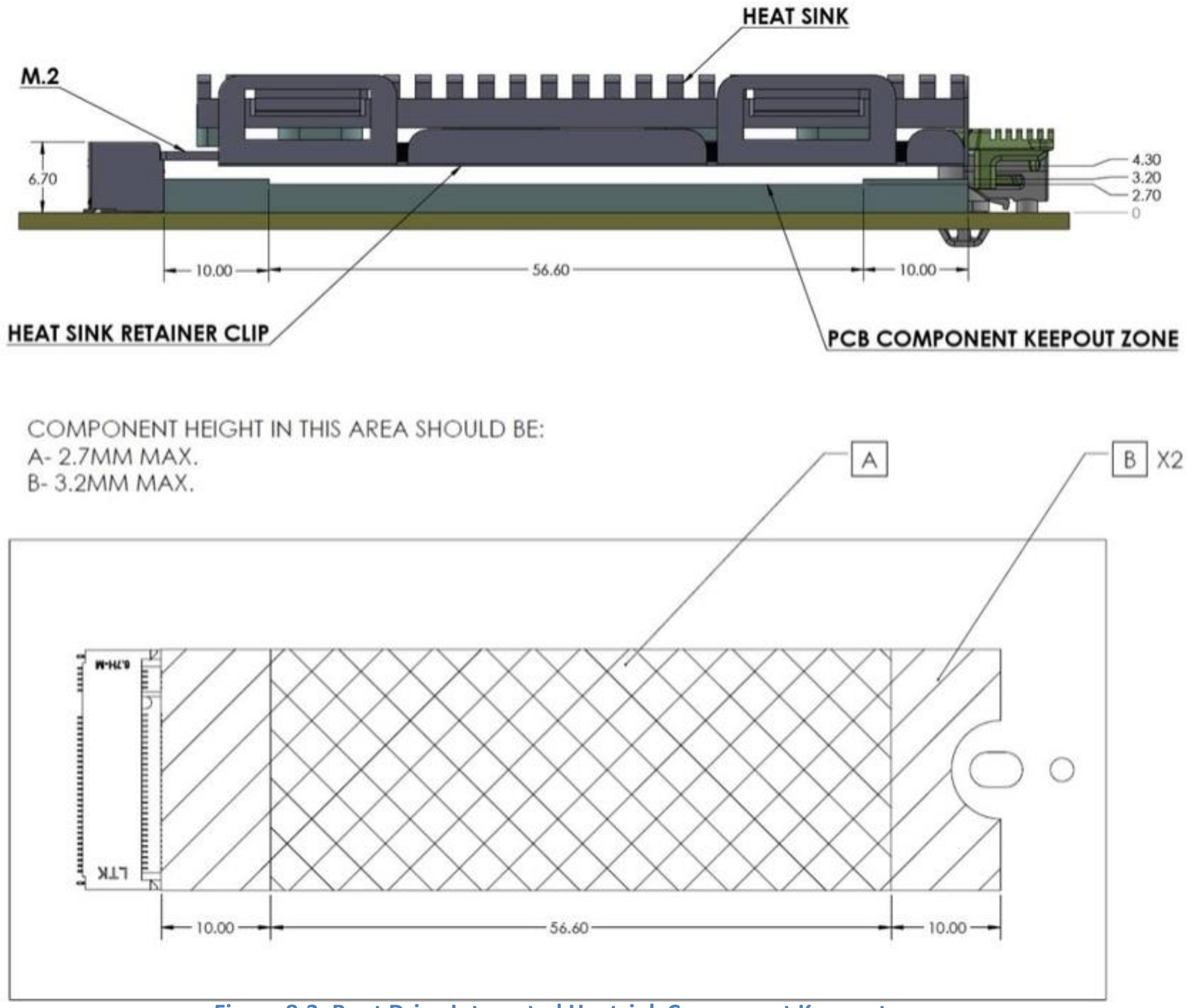


Figure 8-3: Boot Drive Integrated Heatsink Component Keepout

Components under the storage M.2 must follow the keep out requirement as shown in Figure 8-4.

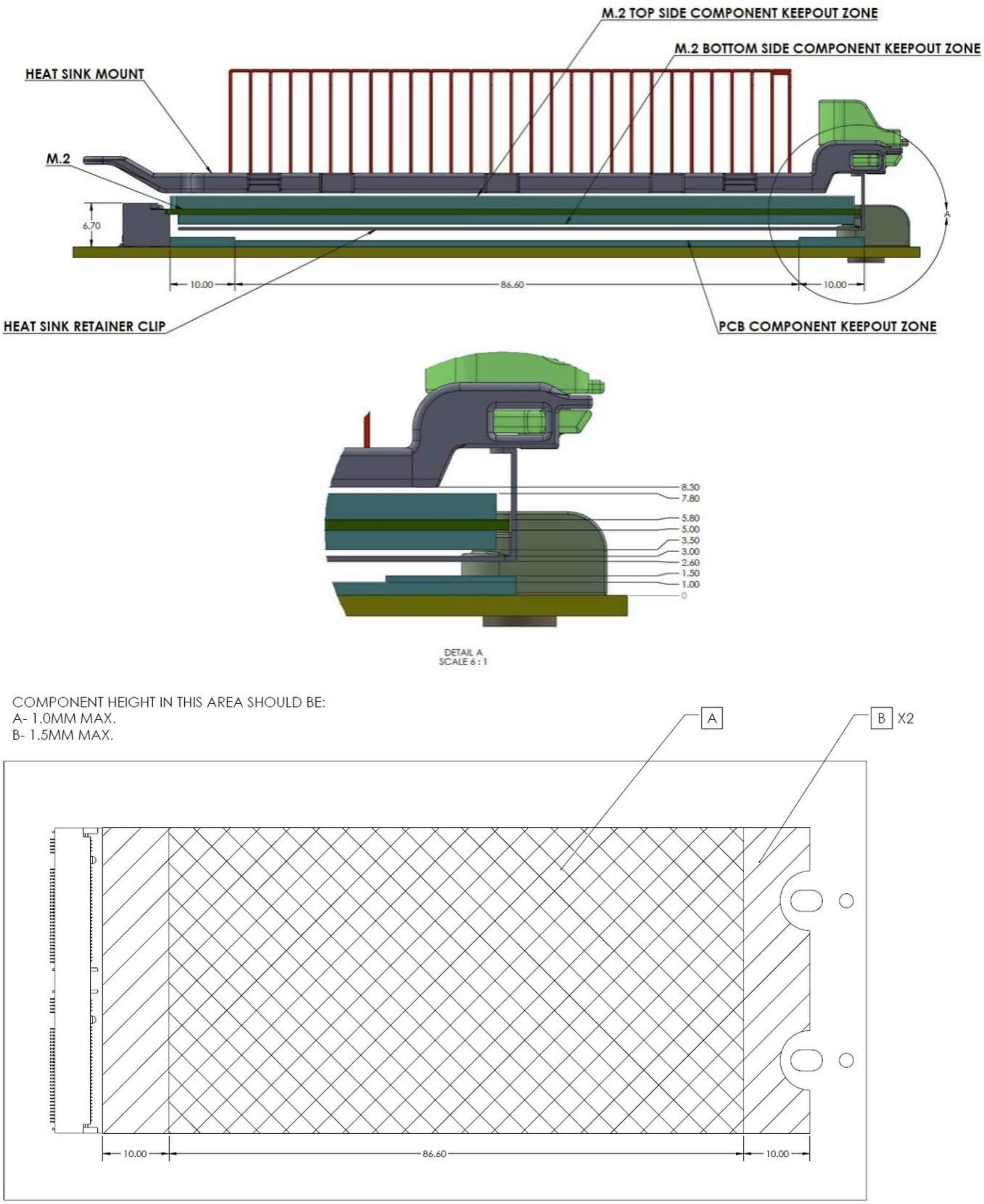


Figure 8-4: Data Drive Integrated Heatsink Component Keepout

8.6 Temperature and Power Sensors

Each card must provide following sensors:

- Temperature sensors for PCH, CPU, DIMM, voltage regulators and other critical chips
- Power sensors for the PCH, CPU and the whole card
- Voltage sensors for all voltage rails
- Power/current sensors for rails providing more than 10W.
- One inlet ambient temperature sensor and one outlet ambient temperature sensor

The BMC on the platform must be able to read all these sensors via the Bridge IC. Additionally, over-temperature thresholds must be configurable, and an alert mechanism must be provided to enable thermal shutdown and/or an increase in airflow. The sensors are accurate to +/-2°C and desired to be within 2% tolerance across whole operation temperature range. The goal sensor accuracy is +/-1°C.

8.7 Server Card Air Baffle

The card level air baffle must be designed to help maintain temperatures of all major components on the server card by reducing bypass air and increasing airflow through key components. The air baffle must be easy to service with the goal of requiring no tooling to remove. The air baffle must not have a large adverse effect on system level pressure drop. Due to height limitation, air baffle is not required for 1U blade form factor but shall be considered for 2U blade form factor.

9 Electrical

9.1 Design Guidelines

Refer to 3rd Gen Intel Xeon Scalable processor CPU and PCH documents for design guidelines.

9.2 Electrical Interfaces of Delta Lake 1S Server Card

9.2.1 Straddle mount connector to YV3 platform Integrated baseboard (with NIC)

A straddle mount 1C connector enables Delta Lake to connect to the platform Integrated baseboard that also has the NIC. A 1:1 cable is used to make the connection between Delta Lake and Integrated baseboard.

The list of signals are as follows:

- Up to 16x lanes (4x for each Delta Lake) of PCIe interfaces with 1x PCIe clock, 1x PERST# output
- 1x IPMB interface for communicate between management devices on Delta Lake (Bridge IC) and Yosemite V3 Baseboard (BMC or Bridge IC)
- 1x I2C Interface (Baseboard as master, for mux controlling and other purpose)
- 1x USB Interface (Baseboard as host for high speed communication with Delta lake BIC and expansion card BIC)

- 1x UART Interface for debug port to BMC
- MISC single ended signals
- Reserved for future use (RSVD)
- Signal names marked with RSVD are for future use. Vendor shall implement these functions with resistor option to isolate it from the interface where needed.

Table 9-1: Delta Lake 1S Server cable stradle mount pin-out (4 lanes)

YV3_pinout_tables			
B1	I2C_IPMB_SDA	BB_BIC_READY	A1
B2	I2C_IPMB_SCL	I2C_BMC_CPLD_SDA	A2
B3	GND	I2C_BMC_CPLD_SCL	A3
B4	AC_ON_OFF_BTN	I2C_BMC_CPLD_ALT_N	A4
B5	HSC_FAULT_N	GND	A5
B6	HSC_EN	PWRBTN_N	A6
B7	STBY_PWROK	RST_BMC_N	A7
B8	PCIE_RESET_N	RSVD	A8
B9	UART_REQ_N	SB_SLOT_ID1	A9
B10	GND	SB_SLOT_ID0	A10
B11	UART_RX	RSVD	A11
B12	UART_TX	MB_PRSNT_N	A12
B13	GND	GND	A13
B14	USB-	REFCLKn0	A14
B15	USB+	REFCLKp0	A15
B16	GND	GND	A16
B17	PETn0	PERn0	A17
B18	PETp0	PERp0	A18
B19	GND	GND	A19
B20	PETn1	PERn1	A20
B21	PETp1	PERp1	A21
B22	GND	GND	A22
B23	PETn2	PERn2	A23
B24	PETp2	PERp2	A24
B25	GND	GND	A25
B26	PETn3	PERn3	A26
B27	PETp3	PERp3	A27
B28	GND	GND	A28

9.2.2 Pin Definitions

The table below provides a detailed description the pins between baseboard and 1S SoC Server Module in Class 1 and Class 2 use cases. Refer to Overview for the context of Class 1 and Class use cases. Vendor shall implement one schematic and CAD, and use BOM option to fulfill Class 1 and Class 2 use cases.

The direction of the signals is always defined from the perspective of the Delta Lake 1S SoC Server module in this specification unless explicitly called out.

Table 9-2: Pin Description of Baseboard to 1S SoC card interface (Class 1)

YV3_pinout_table Baseboard signals (class 1)		
Signal name	**Direction** (In perspective of CPU card)	**Description (Class 1 BMC on Baseboard)**
I2C_IPMB_SDA	I/O	1MHz IPMB between Baseboard management entity (BMC) to BIC on 1S Server Card
I2C_IPMB_SCL	I/O	1MHz IPMB between Baseboard management entity (BMC) to BIC on 1S Server Card
I2C_BMC_CPLD_SDA	I/O	400KHz I2C from Baseboard to CPLD+BIC for commands
I2C_BMC_CPLD_SCL	I/O	400KHz I2C from Baseboard to CPLD+BIC for commands
I2C_BMC_CPLD_ALT_N	Output	Alert signal to baseboard, active low OD signal with PU at Baseboard
MB_PRSNT_N	Output	Present signal. Active low. 1S Server card to place 100 ohm to GND
HSC_EN	Input	Signal to enable Server Card HSC Active High push pull. PD at 1S Server Card side Baseboard assert HSC_EN when: 1) 1S Server Card is fully inserted
HSC_FAULT_N	Output	HSC Fault signal; low active; OD with PU on baseboard.

UART_RX	Input	UART input to 1S Server Card - Source is CPLD on Baseboard - Destination is CPLD on 1S Server Card
UART_TX	Output	UART TX from Server Card - Source is CPLD on 1S Server Card - Destination is CPLD on Baseboard
PCIE_RESET_N	Output	PCIe reset from Server Card to Baseboard Low active, Push-Pull, 3.3V_STBY domain
STBY_PWROK	Output	Standby Power OK of Server Card High active, push-pull
PWRBTN_N	Input	From Baseboard CPLD to CPU card CPLD and BIC to initiate a DC power cycle.
RST_BMC_N	Output	Signal from CPU card BIC to reset BMC on Baseboard
AC_ON_OFF_BTN	Output	AC button on the server module
RSVDx	N/A	Spare signals between server blade and baseboard
REFCLK(n/p)	Output	Server output of 100MHz clock; 1 clock output
PET(n/p)X	Output	PCIe Gen3 RX of Server module. X ranges from 0 to 3 for Delta Lake
PER(n/p)X	Input	PCIe Gen3 RX of Server module. X ranges from 0 to 3
SB_SLOT_ID0/1	Input	Strap on connector chassis showing the Baseboard and Server board, which slot in the sled the server board and sled management cable are connected to
UART_REQ_N	N/A	NC
BB_BIC_READY	Input	Indicates to CPU card CPLD that BMC is ready on Baseboard
USB +/-	I/O	USB interface from BMC on Base board to hub on Delta lake to connect BIC on Delta lake and expansion cards.

Table 9-3: Pin Description of Baseboard to 1S SoC card interface (Class 2)

YV3_pinout_table Baseboard signals (Class 2)

Signal name	**Direction** (In perspective of CPU card)	Description (Class 2, BMC on NIC Expansion Card)
I2C_IPMB_SDA	I/O	1MHz IPMB between Baseboard management entity (BIC) to BIC on 1S Server Card
I2C_IPMB_SCL	I/O	1MHz IPMB between Baseboard management entity (BIC) to BIC on 1S Server Card
I2C_BMC_CPLD_SDA	N/A	NC
I2C_BMC_CPLD_SCL	N/A	NC
I2C_BMC_CPLD_ALT_N	Input	NC
MB_PRSNT_N	Output	Present signal. Active low. 1S Server card to place 100 ohm to GND
HSC_EN	Input	Signal to enable Server Card HSC Active High push pull. PD at 1S Server Card side Baseboard assert HSC_EN when: 1) 1S Server Card is fully inserted
HSC0_FAULT_N	Output	HSC Fault signal; low active; OD with PU at baseboard side.
UART_RX	Input	UART input to 1S Server Card - Source is CPLD on Baseboard - Destination is CPLD on 1S Server Card
UART_TX	Output	UART TX from Server Card - Source is CPLD on 1S Server Card - Destination is CPLD on Baseboard
PCIE_RESET_N	N/A	NC
STBY_PWROK	Output	Standby Power OK of Server Card High active, push-pull
PWRBTN_N	Input	From Baseboard CPLD to CPU card CPLD and BIC to initiate a DC power cycle.
RST_BMC_N	Output	Signal from CPU card BIC to reset BIC on Baseboard
AC_ON_OFF_BTN	Output	AC button on the server module
RSVDx	N/A	Spare signals between server blade and baseboard
REFCLK(n/p)	N/A	NC

PET(n/p)X	N/A	NC
PER(n/p)X	N/A	NC
UART_REQ_N	Input	For Baseboard to request UART channel Input to Server Card to Server Card CPLD Low active, OD, PU at Server Card
BB_BIC_READY	Input	Indicates to CPU card CPLD that BIC is ready on Baseboard
USB +/-	I/O	NC

9.2.3 1U Front Expansion and 2U Front Expansion

The Delta lake server has 2 expansion interfaces.

1U front expansion interface includes:

1. x16 PCIe
2. Reference clocks for devices on expansion
3. SMBUS or I2C interface for sideband management/thermal information
4. Control/alert signals like resets, powerbreak, present etc.
5. Power

An important theme to remember in designing for the expansion system is to ensure that the following are being taken care

1. Side band control and reporting
2. Error logging
3. Device debugging
4. Power delivery from the host system
5. Thermally-cool
6. SI for high speed signals

Table 9-4: 2U expansion pin-out (primary - 4C+)

OA1	P12V_STBY	P12V_STBY	OB1
OA2	P12V_STBY	P12V_STBY	OB2
OA3	P12V_STBY	P12V_STBY	OB3
OA4	P12V_STBY	P12V_STBY	OB4

OA5	P12V_STBY	P12V_STBY	OB5
OA6	P12V_STBY	P12V_STBY	OB6
OA7	P12V_STBY	P12V_STBY	OB7
OA8	P12V_STBY	P12V_STBY	OB8
OA9	P12V_STBY	P12V_STBY	OB9
OA10	NC, RSVD	NC, RSVD	OB10
OA11	GND	P3V3_STBY	OB11
OA12	GND	NC, RSVD	OB12
OA13	GND	GND	OB13
OA14	GND	GND	OB14
Mechanical key			
A1	GND	GND	B1
A2	GND	GND	B2
A3	GND	GND	B3
A4	GND	GND	B4
A5	GND	GND	B5
A6	GND	GND	B6
A7	GND	GND	B7
A8	FM_RISER_MAIN_PWR_R_EN	FM_PCIE_RISER_PWRBRK_N	B8
A9	PWRGD_RISER_MAIN_PWR	RISER_CARD_TYPE_DETECTION0	B9
A10	PRSNTA1_N	RISER_CARD_TYPE_DETECTION1	B10
A11	RST_PCIE_RISER_PERST1_N	RISER_CARD_TYPE_DETECTION2	B11
A12	RISER_RSVD1	FM_RISER_STBY_PWR_R_EN	B12
A13	GND	GND	B13
A14	SMB_PCIE_RISER_SCL	CLK_100M_RISER_PCIE_CLK0_DN	B14
A15	SMB_PCIE_RISER_SDA	CLK_100M_RISER_PCIE_CLK0_DP	B15
A16	GND	GND	B16
A17	NC	NC	B17
A18	NC	NC	B18
A19	GND	GND	B19
A20	NC	NC	B20
A21	NC	NC	B21
A22	GND	GND	B22
A23	NC	NC	B23
A24	NC	NC	B24
A25	GND	GND	B25
A26	NC	NC	B26
A27	NC	NC	B27
A28	GND	GND	B28

Mechanical key			
A29	GND	GND	B29
A30	NC	NC	B30
A31	NC	NC	B31
A32	GND	GND	B32
A33	NC	NC	B33
A34	NC	NC	B34
A35	GND	GND	B35
A36	NC	NC	B36
A37	NC	NC	B37
A38	GND	GND	B38
A39	NC	NC	B39
A40	NC	NC	B40
A41	GND	GND	B41
A42	NC	NC	B42
Mechanical key			
A43	GND	GND	B43
A44	P3E_CPU_RISER_RX_DN_7	P3E_CPU_RISER_TX_C_DN_7	B44
A45	P3E_CPU_RISER_RX_DP_7	P3E_CPU_RISER_TX_C_DP_7	B45
A46	GND	GND	B46
A47	P3E_CPU_RISER_RX_DN_6	P3E_CPU_RISER_TX_C_DN_6	B47
A48	P3E_CPU_RISER_RX_DP_6	P3E_CPU_RISER_TX_C_DP_6	B48
A49	GND	GND	B49
A50	P3E_CPU_RISER_RX_DN_5	P3E_CPU_RISER_TX_C_DN_5	B50
A51	P3E_CPU_RISER_RX_DP_5	P3E_CPU_RISER_TX_C_DP_5	B51
A52	GND	GND	B52
A53	P3E_CPU_RISER_RX_DN_4	P3E_CPU_RISER_TX_C_DN_4	B53
A54	P3E_CPU_RISER_RX_DP_4	P3E_CPU_RISER_TX_C_DP_4	B54
A55	GND	GND	B55
A56	P3E_CPU_RISER_RX_DN_3	P3E_CPU_RISER_TX_C_DN_3	B56
A57	P3E_CPU_RISER_RX_DP_3	P3E_CPU_RISER_TX_C_DP_3	B57
A58	GND	GND	B58
A59	P3E_CPU_RISER_RX_DN_2	P3E_CPU_RISER_TX_C_DN_2	B59
A60	P3E_CPU_RISER_RX_DP_2	P3E_CPU_RISER_TX_C_DP_2	B60
A61	GND	GND	B61
A62	P3E_CPU_RISER_RX_DN_1	P3E_CPU_RISER_TX_C_DN_1	B62
A63	P3E_CPU_RISER_RX_DP_1	P3E_CPU_RISER_TX_C_DP_1	B63
A64	GND	GND	B64
A65	P3E_CPU_RISER_RX_DN_0	P3E_CPU_RISER_TX_C_DN_0	B65

A66	P3E_CPU_RISER_RX_DP_0	P3E_CPU_RISER_TX_C_DP_0	B66
A67	GND	GND	B67
A68	USB2_RISER_DN	UART_RISER_RX	B68
A69	USB2_RISER_DP	UART_RISER_TX	B69
A70	GND	PRSNTA1_N	B70

Table 9-5: 2U expansion pin-out (secondary - 4C+)

OA1	P12V_STBY	P12V_STBY	OB1
OA2	P12V_STBY	P12V_STBY	OB2
OA3	P12V_STBY	P12V_STBY	OB3
OA4	P12V_STBY	P12V_STBY	OB4
OA5	P12V_STBY	P12V_STBY	OB5
OA6	P12V_STBY	P12V_STBY	OB6
OA7	P12V_STBY	P12V_STBY	OB7
OA8	P12V_STBY	P12V_STBY	OB8
OA9	P12V_STBY	P12V_STBY	OB9
OA10	NC, RSVD	NC, RSVD	OB10
OA11	GND	P3V3_STBY	OB11
OA12	GND	NC, RSVD	OB12
OA13	GND	GND	OB13
OA14	GND	GND	OB14
Mechanical key			
A1	GND	GND	B1
A2	GND	GND	B2
A3	GND	GND	B3
A4	GND	GND	B4
A5	GND	GND	B5
A6	GND	GND	B6
A7	GND	GND	B7
A8	SMB_RISER_ALT_N	RST_PCIE_RISER_PERST_N	B8
A9	RISER_RSVD2	RISER_LOAD_R	B9
A10	PRSNTA2_N	RISER_DATA_IN	B10
A11	RISER_RSVD3	RISER_DATA_OUT_R	B11
A12	RISER_RSVD4	RISER_CLK_R	B12
A13	GND	GND	B13
A14	IPMB_PCIE_RISER_SCL	CLK_100M_RISER_PCIE_CLK1_DN	B14
A15	IPMB_PCIE_RISER_SDA	CLK_100M_RISER_PCIE_CLK1_DP	B15
A16	GND	GND	B16

A17	P3E_CPU_RISER_RX_DN_31	P3E_CPU_RISER_TX_C_DN_31	B17
A18	P3E_CPU_RISER_RX_DP_31	P3E_CPU_RISER_TX_C_DP_31	B18
A19	GND	GND	B19
A20	P3E_CPU_RISER_RX_DN_30	P3E_CPU_RISER_TX_C_DN_30	B20
A21	P3E_CPU_RISER_RX_DP_30	P3E_CPU_RISER_TX_C_DP_30	B21
A22	GND	GND	B22
A23	P3E_CPU_RISER_RX_DN_29	P3E_CPU_RISER_TX_C_DN_29	B23
A24	P3E_CPU_RISER_RX_DP_29	P3E_CPU_RISER_TX_C_DP_29	B24
A25	GND	GND	B25
A26	P3E_CPU_RISER_RX_DN_28	P3E_CPU_RISER_TX_C_DN_28	B26
A27	P3E_CPU_RISER_RX_DP_28	P3E_CPU_RISER_TX_C_DP_28	B27
A28	GND	GND	B28
Mechanical key			
A29	GND	GND	B29
A30	P3E_CPU_RISER_RX_DN_27	P3E_CPU_RISER_TX_C_DN_27	B30
A31	P3E_CPU_RISER_RX_DP_27	P3E_CPU_RISER_TX_C_DP_27	B31
A32	GND	GND	B32
A33	P3E_CPU_RISER_RX_DN_26	P3E_CPU_RISER_TX_C_DN_26	B33
A34	P3E_CPU_RISER_RX_DP_26	P3E_CPU_RISER_TX_C_DP_26	B34
A35	GND	GND	B35
A36	P3E_CPU_RISER_RX_DN_25	P3E_CPU_RISER_TX_C_DN_25	B36
A37	P3E_CPU_RISER_RX_DP_25	P3E_CPU_RISER_TX_C_DP_25	B37
A38	GND	GND	B38
A39	P3E_CPU_RISER_RX_DN_24	P3E_CPU_RISER_TX_C_DN_24	B39
A40	P3E_CPU_RISER_RX_DP_24	P3E_CPU_RISER_TX_C_DP_24	B40
A41	GND	GND	B41
A42	NC	NC	B42
Mechanical key			
A43	GND	GND	B43
A44	P3E_CPU_RISER_RX_DN_23	P3E_CPU_RISER_TX_C_DN_23	B44
A45	P3E_CPU_RISER_RX_DP_23	P3E_CPU_RISER_TX_C_DP_23	B45
A46	GND	GND	B46
A47	P3E_CPU_RISER_RX_DN_22	P3E_CPU_RISER_TX_C_DN_22	B47
A48	P3E_CPU_RISER_RX_DP_22	P3E_CPU_RISER_TX_C_DP_22	B48
A49	GND	GND	B49
A50	P3E_CPU_RISER_RX_DN_21	P3E_CPU_RISER_TX_C_DN_21	B50
A51	P3E_CPU_RISER_RX_DP_21	P3E_CPU_RISER_TX_C_DP_21	B51
A52	GND	GND	B52
A53	P3E_CPU_RISER_RX_DN_20	P3E_CPU_RISER_TX_C_DN_20	B53

A54	P3E_CPU_RISER_RX_DP_20	P3E_CPU_RISER_TX_C_DP_20	B54
A55	GND	GND	B55
A56	P3E_CPU_RISER_RX_DN_19	P3E_CPU_RISER_TX_C_DN_19	B56
A57	P3E_CPU_RISER_RX_DP_19	P3E_CPU_RISER_TX_C_DP_19	B57
A58	GND	GND	B58
A59	P3E_CPU_RISER_RX_DN_18	P3E_CPU_RISER_TX_C_DN_18	B59
A60	P3E_CPU_RISER_RX_DP_18	P3E_CPU_RISER_TX_C_DP_18	B60
A61	GND	GND	B61
A62	P3E_CPU_RISER_RX_DN_17	P3E_CPU_RISER_TX_C_DN_17	B62
A63	P3E_CPU_RISER_RX_DP_17	P3E_CPU_RISER_TX_C_DP_17	B63
A64	GND	GND	B64
A65	P3E_CPU_RISER_RX_DN_16	P3E_CPU_RISER_TX_C_DN_16	B65
A66	P3E_CPU_RISER_RX_DP_16	P3E_CPU_RISER_TX_C_DP_16	B66
A67	GND	GND	B67
A68	RSVD_USB_DN	RSVD_UART_RX	B68
A69	RSVD_USB_DP	RSVD_UART_TX	B69
A70	GND	PRSNTA2_N	B70

Table 9-6: Front expansion pin-out (primary - 4C+)

Side B		Side A	
OCP_B1	NIC_PWR_GOOD	PERST2#	OCP_A1
OCP_B2	MAIN_PWR_EN	PERST3#	OCP_A2
OCP_B3	LD#	WAKE_N	OCP_A3
OCP_B4	DATA_IN	RFU, NC	OCP_A4
OCP_B5	DATA_OUT	RFU, NC	OCP_A5
OCP_B6	CLK	SLOT_ID1	OCP_A6
OCP_B7	SLOT_ID0	RFU, NC	OCP_A7
OCP_B8	RFU, NC	RFU, NC	OCP_A8
OCP_B9	RFU, NC	RFU, NC	OCP_A9
OCP_B10	GND	GND	OCP_A10
OCP_B11	SMCLK1	SM_ALT_N	OCP_A11
OCP_B12	SMDAT1	CARD_TYPE_EXP	OCP_A12
OCP_B13	GND	GND	OCP_A13
OCP_B14	RFU, NC	RFU, NC	OCP_A14

Mechanical Key			
B1	+12V_EDGE	GND	A1
B2	+12V_EDGE	GND	A2
B3	+12V_EDGE	GND	A3
B4	+12V_EDGE	GND	A4
B5	+12V_EDGE	GND	A5
B6	+12V_EDGE	GND	A6
B7	RFU, NC	SMCLK2	A7
B8	RFU, NC	SMDAT2	A8
B9	RFU, NC	RST_BMC_N	A9
B10	PERST0#	PRSNTA#	A10
B11	+3.3V_EDGE	PERST1#	A11
B12	AUX_PWR_EN	PRSNTB2#	A12
B13	GND	GND	A13
B14	REFCLKn0	UART_PR_RX	A14
B15	REFCLKp0	UART_PR_TX	A15
B16	GND	GND	A16
B17	PETn0	PERn0	A17
B18	PETp0	PERp0	A18
B19	GND	GND	A19
B20	PETn1	PERn1	A20
B21	PETp1	PERp1	A21
B22	GND	GND	A22
B23	PETn2	PERn2	A23
B24	PETp2	PERp2	A24
B25	GND	GND	A25
B26	PETn3	PERn3	A26
B27	PETp3	PERp3	A27
B28	GND	GND	A28
Mechanical Key			
B29	GND	GND	A29
B30	PETn4	PERn4	A30
B31	PETp4	PERp4	A31
B32	GND	GND	A32
B33	PETn5	PERn5	A33
B34	PETp5	PERp5	A34
B35	GND	GND	A35

B36	PETn6	PERn6	A36
B37	PETp6	PERp6	A37
B38	GND	GND	A38
B39	PETn7	PERn7	A39
B40	PETp7	PERp7	A40
B41	GND	GND	A41
B42	PRSNTB0#	PRSNTB1#	A42
Mechanical Key			
B43	GND	GND	A43
B44	PETn8	PERn8	A44
B45	PETp8	PERp8	A45
B46	GND	GND	A46
B47	PETn9	PERn9	A47
B48	PETp9	PERp9	A48
B49	GND	GND	A49
B50	PETn10	PERn10	A50
B51	PETp10	PERp10	A51
B52	GND	GND	A52
B53	PETn11	PERn11	A53
B54	PETp11	PERp11	A54
B55	GND	GND	A55
B56	PETn12	PERn12	A56
B57	PETp12	PERp12	A57
B58	GND	GND	A58
B59	PETn13	PERn13	A59
B60	PETp13	PERp13	A60
B61	GND	GND	A61
B62	PETn14	PERn14	A62
B63	PETp14	PERp14	A63
B64	GND	GND	A64
B65	PETn15	PERn15	A65
B66	PETp15	PERp15	A66
B67	GND	GND	A67
B68	UART_NIC_BMC_TX	USB2_EXP_N	A68
B69	UART_NIC_BMC_RX	USB2_EXP_P	A69
B70	PRSNTB3#	PWRBRK0#	A70

Table 9-7: Front expansion pin-out (secondary, 4C)

B1	+12V_EDGE	GND	A1
B2	+12V_EDGE	GND	A2
B3	+12V_EDGE	GND	A3
B4	+12V_EDGE	GND	A4
B5	+12V_EDGE	GND	A5
B6	+12V_EDGE	GND	A6
B7	RFU, NC	RFU, NC	A7
B8	RFU, NC	RFU, NC	A8
B9	RFU, NC	RFU, NC	A9
B10	RFU, NC	RFU, NC	A10
B11	+3.3V_EDGE	RFU, NC	A11
B12	RFU, NC	RFU, NC	A12
B13	GND	GND	A13
B14	RFU, NC	RFU, NC	A14
B15	RFU, NC	RFU, NC	A15
B16	GND	GND	A16
B17	RFU, NC	RFU, NC	A17
B18	RFU, NC	RFU, NC	A18
B19	GND	GND	A19
B20	RFU, NC	RFU, NC	A20
B21	RFU, NC	RFU, NC	A21
B22	GND	GND	A22
B23	RFU, NC	RFU, NC	A23
B24	RFU, NC	RFU, NC	A24
B25	GND	GND	A25
B26	RFU, NC	RFU, NC	A26
B27	RFU, NC	RFU, NC	A27
B28	GND	GND	A28
Mechanical Key			
B29	GND	GND	A29
B30	RFU, NC	RFU, NC	A30
B31	RFU, NC	RFU, NC	A31
B32	GND	GND	A32
B33	RFU, NC	RFU, NC	A33
B34	RFU, NC	RFU, NC	A34
B35	GND	GND	A35

B36	RFU, NC	RFU, NC	A36
B37	RFU, NC	RFU, NC	A37
B38	GND	GND	A38
B39	RFU, NC	RFU, NC	A39
B40	RFU, NC	RFU, NC	A40
B41	GND	GND	A41
B42	RFU, NC	RFU, NC	A42
Mechanical Key			
B43	GND	GND	A43
B44	RFU, NC	RFU, NC	A44
B45	RFU, NC	RFU, NC	A45
B46	GND	GND	A46
B47	RFU, NC	RFU, NC	A47
B48	RFU, NC	RFU, NC	A48
B49	GND	GND	A49
B50	RFU, NC	RFU, NC	A50
B51	RFU, NC	RFU, NC	A51
B52	GND	GND	A52
B53	RFU, NC	RFU, NC	A53
B54	RFU, NC	RFU, NC	A54
B55	GND	GND	A55
B56	RFU, NC	RFU, NC	A56
B57	RFU, NC	RFU, NC	A57
B58	GND	GND	A58
B59	RFU, NC	RFU, NC	A59
B60	RFU, NC	RFU, NC	A60
B61	GND	GND	A61
B62	RFU, NC	RFU, NC	A62
B63	RFU, NC	RFU, NC	A63
B64	GND	GND	A64
B65	RFU, NC	RFU, NC	A65
B66	RFU, NC	RFU, NC	A66
B67	GND	GND	A67
B68	RFU, NC	RFU, NC	A68
B69	RFU, NC	RFU, NC	A69
B70	RFU, NC	RFU, NC	A70

9.2.4 Pin Definitions

As the connector is referenced to OCP NIC 3.0 interface (http://files.opencompute.org/oc/public.php?service=files&t=ea72c1c3050057655da8eda1f42dd0e2&download&path=//OCP_NIC_3.0_draft_0v86_20190402d_TN_THERMAL_TIERS_V1.pdf), this specification will only call out pin definitions that COULD be different for the expansion system

Table 9-8:Detailed Pin Definitions

Signal name	Direction (with respect to server blade)	Description
CARD_TYPE_N	I	Found in front expansion module. Used to determine the card type. 1.5V: Expansion with NIC 1 V: Expansion with 4 EDSFF 0.5V: Expansion with 6 M.2 0V: Direct connect for 1 EDSFF
CARD_TYPE_Detection[2:0]	I	Found in 2U expansion To identify which type of card for the riser 000 – GPV3 with BCM PCIe Switch 001 – 2OU Expansion with no PCIe Switch 010 – Sierra Point 011 – GPV3 with Microchip PCIe Switch 100, 101, 110, 111 -> RFU

9.3 Electrical Interface in Delta Lake 1S Server Card

9.3.1 UART

The serial port shall be routed to the BMC on the platform. Thus, the user can access the system' s serial console through the BMC locally or remotely via Serial Over Lan (SOL).

9.3.2 I^{2}C and IPMB

There are 2x I2C connections from the server to Baseboard as shown in the figure below

I2C_BMC_CPLD connects the BMC to BIC and CPLD. This channel is used for primary for configuration interface. It shall support at least 400kHz while aspired to operate at 1MHz. The BMC will use this channel for command and control of BIC and CPLD in configurations.

I2C for Command will have BIC and BMC connected where its usage is generic IPMB transactions. The channel here is similar to the implementation in Yosemite v2.

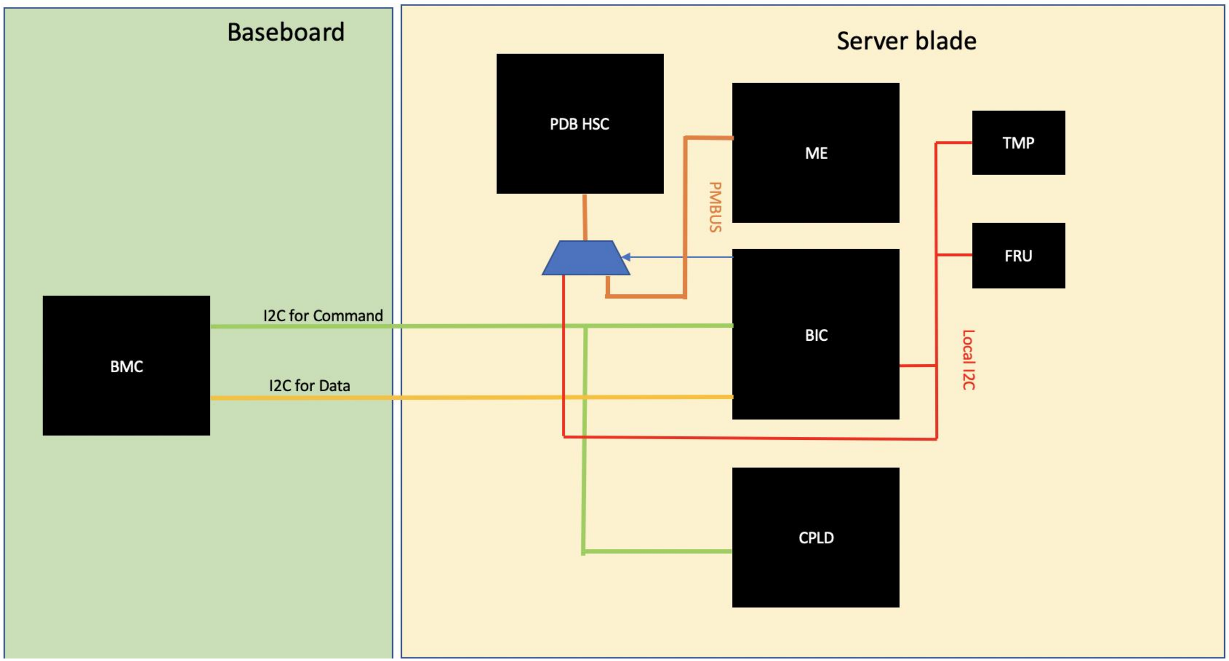


Figure 9-1: I2C connections out of server blade.

9.3.3 USB

A high-speed USB connection is provisioned from BMC to the Delta lake with BMC as host and BIC as device. There a USB hub on Delta lake to enable BMC to connect to B on expansion cards as well as to CPLD via a USB to SPI bridge as shown in Figure 9-2. The USB interface is used as a higher speed interface to perform firmware updates. This reduces the total time required to perform these updates.

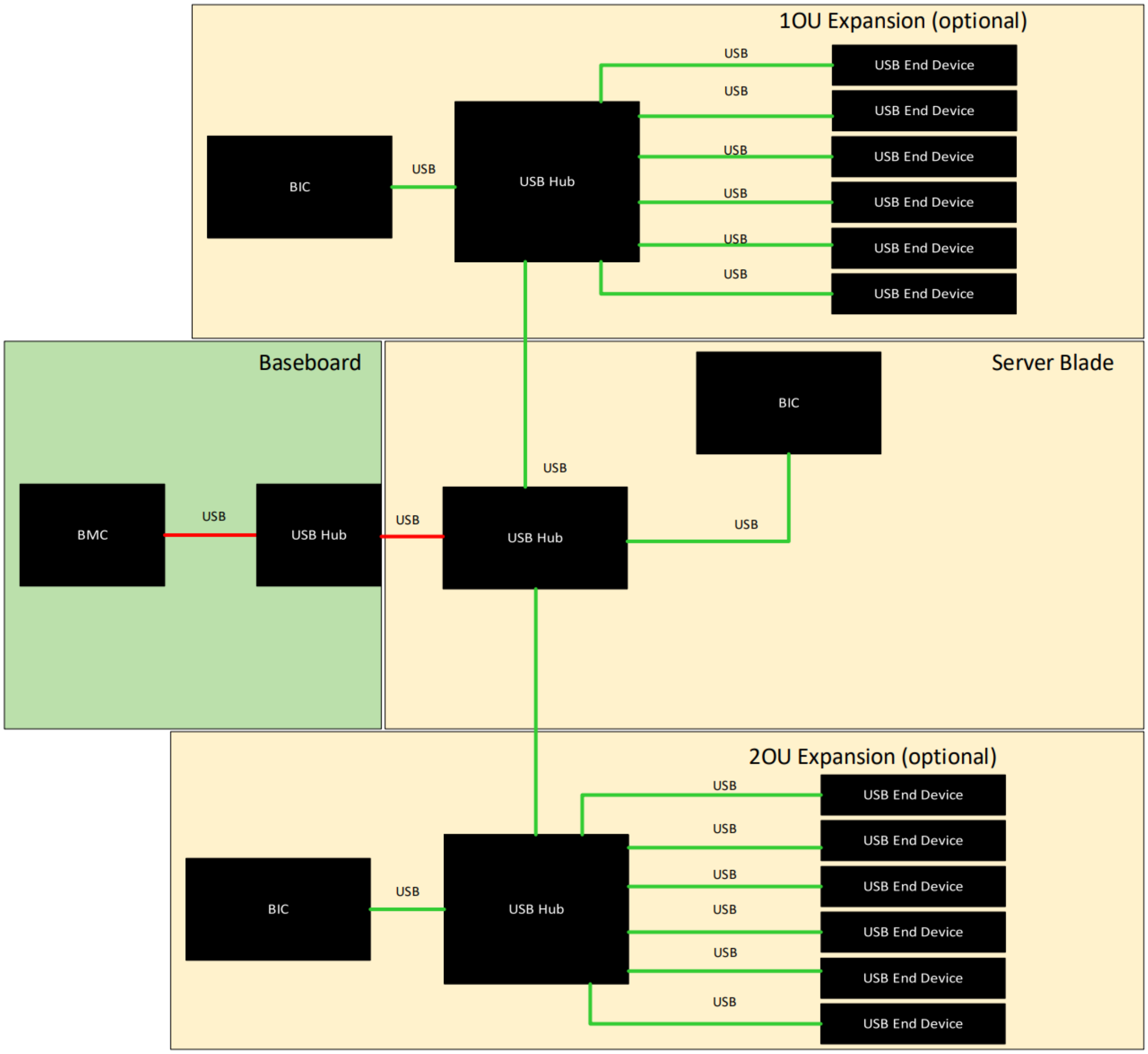


Figure 9-2: USB Connections for Class 1

In Class 2 configuration with local NIC, the BMC moves from the Baseboard to the NIC expansion board and USB is changed accordingly with BOM options. This can be seen in figure 9-3.

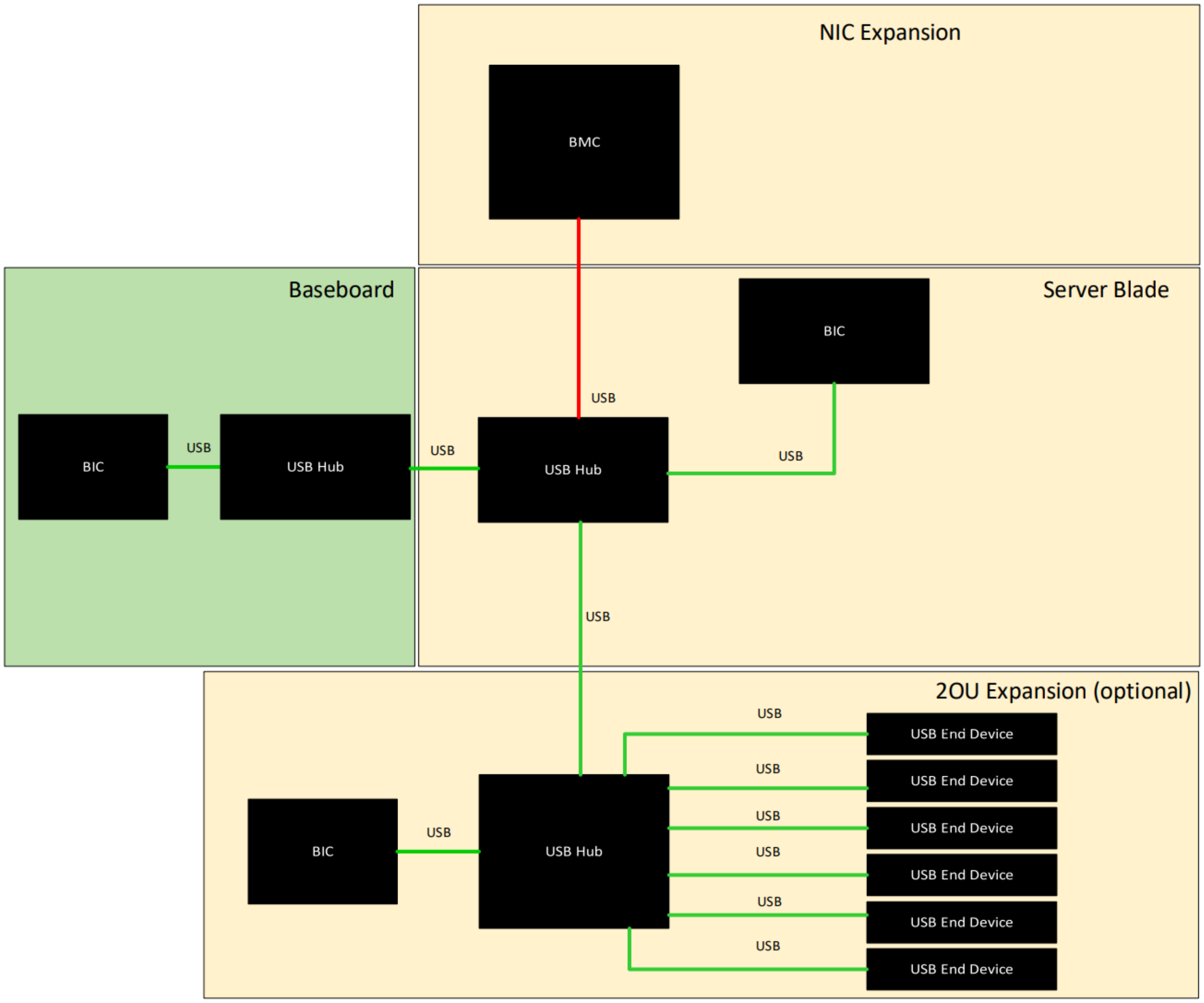


Figure 9-3: USB Connections for Class 2

9.3.4 SPI

SPI is the BIOs interface for PCH. Delta lake uses Quad SPI to load BIOs quickly. PFR uses the CPLD to act as the Platform Root of Trust for ensuring security and preventing intrusion and corruption of SPI Flash. The CPLD can talk to the SPI Flash via the SPI mux which it controls. The CPLD also has a direct path to the SPI Flash to monitor traffic and enforce SPI filtering.

BMC can program SPI Flash Out of Band (OOB), via the BIC. To enable that, there is a mux in the CPLD that allows BIC to use the PFR's SPI path as shown in Figure 9-4 – the performance of the OOB interface will be tested in POC board.

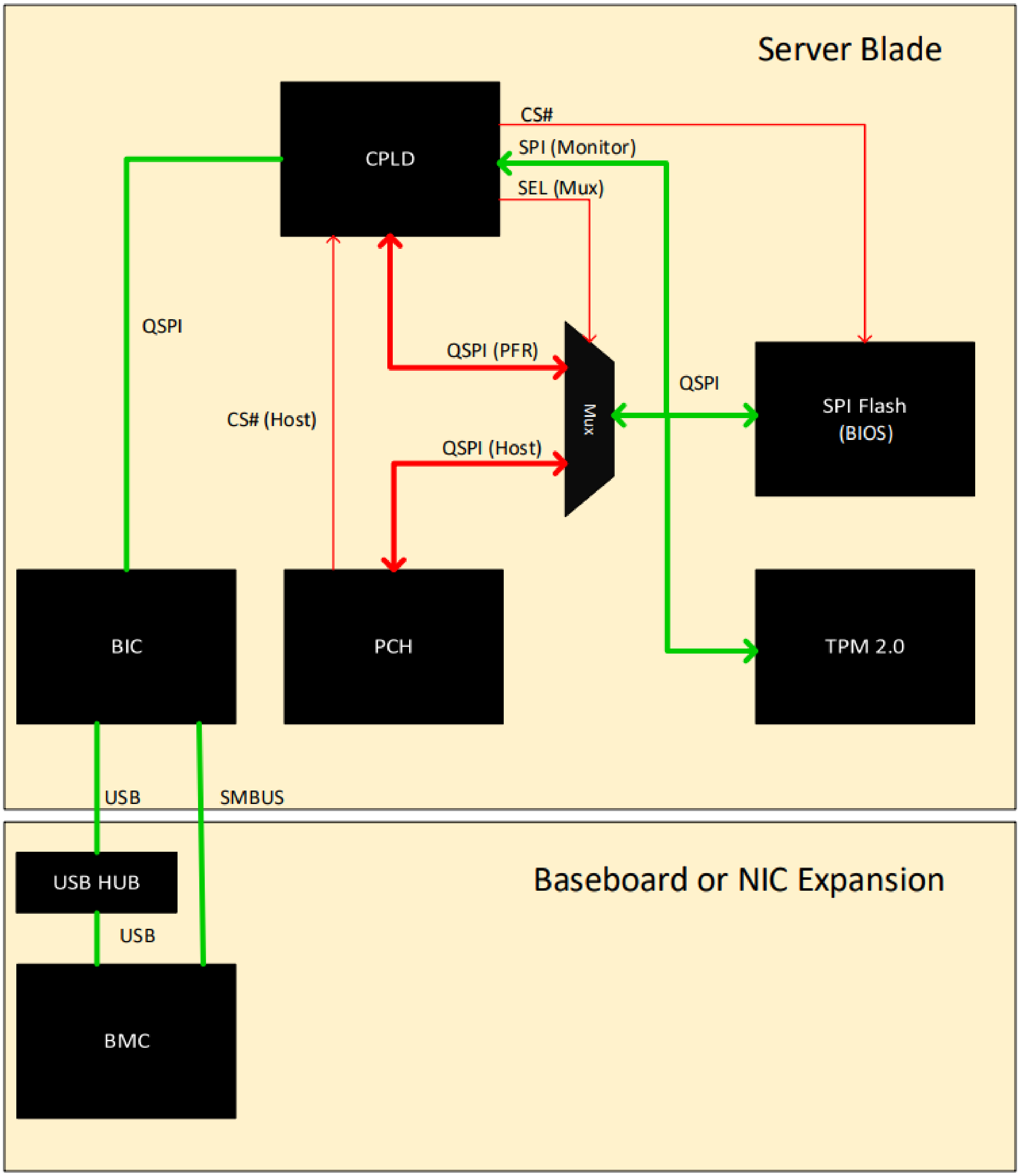


Figure 9-4: SPI Connections for Delta Lake

9.3.5 JTAG

XDP is the primary JTAG port to Delta Lake server board to communicate to PCH and CPU. The BIC is also on the bus and can take over the bus to become the master.

There is dedicated connector to program the CPLD.

A third JTAG connector is available to talk to BIC. CPLD is also on this bus and can take over the bus to talk to BIC.

Additionally, the BMC through the BIC, supports Intel's At-Scale Debug (ASD) to the CPU. In order to support security requirements, this ASD connection can be manually disabled by populating a jumper on Delta Lake. This way, the JTAG interface is only accessible if someone has physical access to the system. This is outlined in figured 9-5.

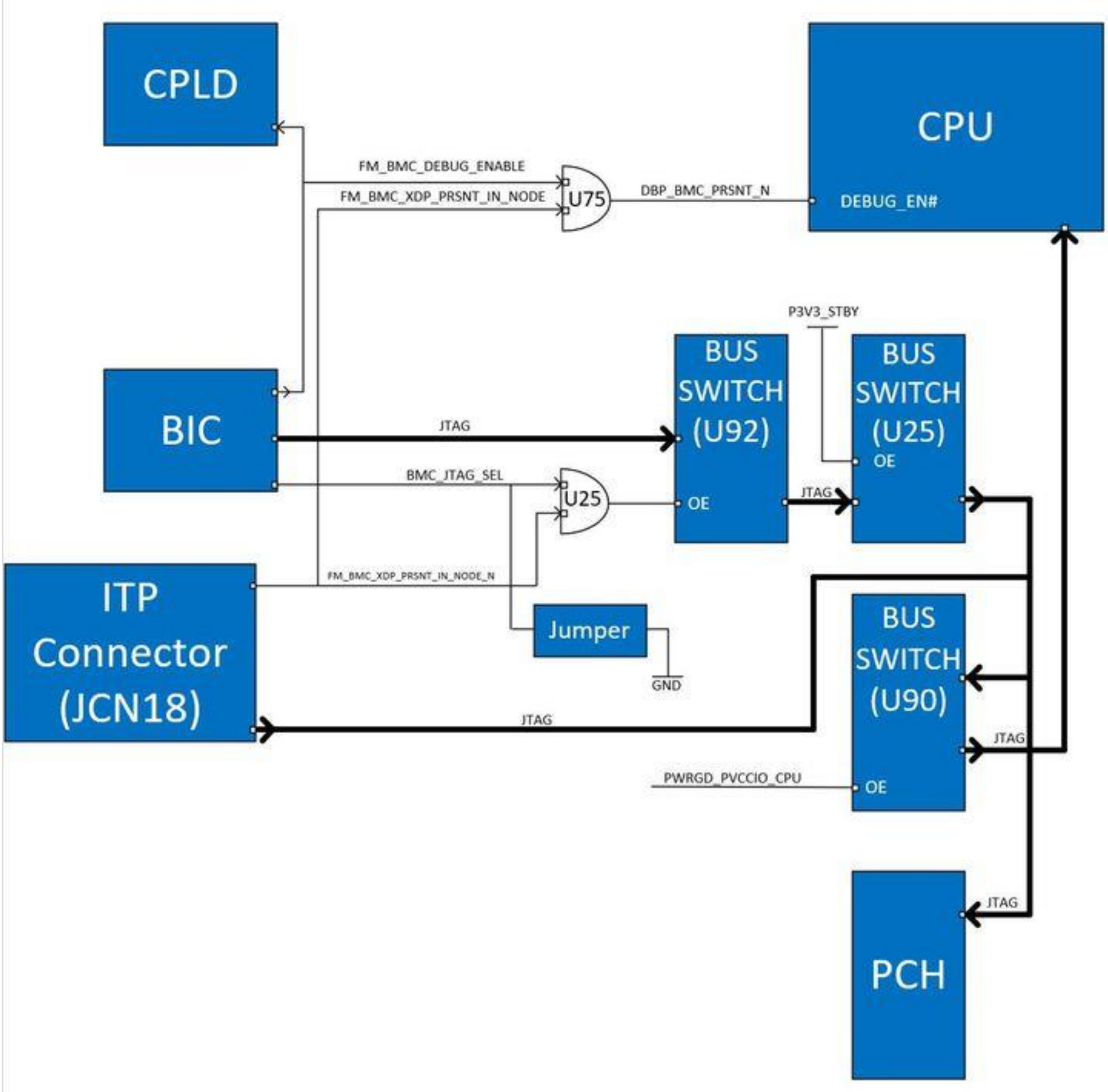


Figure 9-5: At-Scale Debug Connections for Delta Lake

9.3.6 Bridge IC(BIC) GPIOs

The table below lists the GPIO assignments for BIC.

Table 9-9: BIC GPIOs

GPIO	Function	Signal Name	Push-pull/Open drain
PA0	U0RX	SMB_BMC_SPD_ACCESS_STBY_LVC3_SCL	OPEN DRAIN
PA1	U0TX	SMB_BMC_SPD_ACCESS_STBY_LVC3_SDA	OPEN DRAIN
PA2	SSI0CLK	PWRGD_BMC_PS_PWROK_R	
PA3	SSI0FSS	FM_PCH_BMC_THERMTRIP_N	
PA4	SSI0XDAT0	IRQ_HSC_ALERT2_N	
PA5	SSI0XDAT1	FM_BMC_ONCTL_R_N	OPEN DRAIN
PA6	EN0RXCK	RST_RSTBTN_OUT_N	
PA7	USB0EPEN	FM_JTAG_TCK_MUX_SEL	PUSH PULL
PB0	USB0ID	UART_HOST_R_TXD	
PB1	USB0VBUS	UART_HOST_RXD	PUSH PULL
PB2	I2C0SCL	SMB_CPU_PIROM_R_SCL	OPEN DRAIN
PB3	I2C0SDA	SMB_CPU_PIROM_R_SDA	OPEN DRAIN
PB4	I2C5SCL	SMB_SENSOR_LVC3_R_SCL	OPEN DRAIN
PB5	I2C5SDA	SMB_SENSOR_LVC3_R_SDA	OPEN DRAIN
PB6	I2C6SCL	SMB_SMLINK0_STBY_LVC3_R_SCL	OPEN DRAIN
PB7	I2C6SDA	SMB_SMLINK0_STBY_LVC3_R_SDA	OPEN DRAIN
PC0	TCK	JTAG_BIC_TCK	
PC1	TMS	JTAG_BIC_TMS	
PC2	TDI	JTAG_BIC_TDI	
PC3	TDO	JTAG_BIC_TDO	PUSH PULL
PC4	FAN0PWM0	FM_HSC_TIMER	
PC5	FAN0TACH0	FM_CPU_MEMHOT_OUT_N	
PC6	FAN0PWM1	FM_SPD_DDRCPU_LVLSHFT_EN	OPEN DRAIN
PC7	FAN0TACH1	RST_PLTRST_FROM_PCH_N	
PD0	I2C7SCL	SMB_HOST_STBY_BMC_LVC3_R_SCL	OPEN DRAIN
PD1	I2C7SDA	SMB_HOST_STBY_BMC_LVC3_R_SDA	OPEN DRAIN
PD2	I2C8SCL	SMB_PMBUS_SML2_STBY_LVC3_R2_SCL	OPEN DRAIN
PD3	I2C8SDA	SMB_PMBUS_SML2_STBY_LVC3_R2_SDA	OPEN DRAIN
PD4	AIN7		
PD5	AIN6	CARD_TYPE_EXP	
PD6	AIN5	A_PVNN_PCH_STBY_SENSOR	
PD7	AIN4	A_P3V_BAT_SCALED	
PE0	AIN3	A_P1V05_PCH_STBY_SENSOR	
PE1	AIN2	A_P3V3_STBY_SCALED	
PE2	AIN1	A_P3V3_SCALED	
PE3	AIN0	A_P12V_STBY_SCALED	
PE4	FAN0TACH7	IRQ_BMC_PRDY_NODE_OD_N	
PE5	FAN0PWM7	FM_CPU_SKTOCC_LVT3_N	
PE6	SGPMO	SGPIO_BMC_DOUT_R	PUSH PULL
PE7	NMI	IRQ_NMI_EVENT_R_N	

PF0	FAN0PWM2	IRQ_SMB_IO_LVC3_STBY_ALRT_N	
PF1	FAN0TACH2		
PF2	RMII0-EN0MDC	RST_USB_HUB_N	OPEN DRAIN
PF3	RMII0-EN0MDIO	A_P3V_BAT_SCALED_EN	PUSH PULL
PF4	FAN0PWM4	FM_MP_PS_REDUNDANT_LOST_N	
PF5	FAN0TACH4		
PF6	FAN0PWM5	FM_MP_PS_FAIL_N	
PF7	FAN0TACH5	RST_MCP2210_N	OPEN DRAIN
PG0	FAN0PWM6	FM_CPU_THERMTRIP_LATCH_LVT3_N	
PG1	FAN0TACH6	FM_BMC_PCHIE_N	OPEN DRAIN
PG2	EN0TXCK	FM_SLPS4_R_N	
PG3	RMII0-EN0TXEN	JTAG_BMC_TDI_R	PUSH PULL
PG4	RMII0-EN0TXD0	JTAG_BMC_TMS_R	PUSH PULL
PG5	RMII0-EN0TXD1	JTAG_BMC_TDO_R	
PG6	RMII0-EN0RXER	JTAG_BMC_TCK_R	PUSH PULL
PG7	RMII0-EN0RXDV	JTAG_BMC_NTRST_R_N	PUSH PULL
PH0	LPC0COMXRTS	FM_SOL_UART_CH_SEL	PUSH PULL
PH1	LPC0COMXCTS		
PH2	LPC0COMXDCD	FM_BMC_PCH_SCI_LPC_R_N	OPEN DRAIN
PH3	LPC0COMXDSR		
PH4	LPC0COMXDTR		
PH5	LPC0COMXRI		
PH6	GPIOPH6	FM_BMC_DEBUG_ENABLE_N	OPEN DRAIN
PH7	GPIOPH7	DBP_PRESENT_R2_N	
PJ0	PECIVTT	PVCCIO_CPU	
PJ1	PECI0SD	PECI_BMC_R	
PJ2	GPIOPJ2	FM_FAST_PROCHOT_EN_N	PUSH PULL
PJ3	GPIOPJ3	FM_CPU_FIVR_FAULT_LVT3_N	
PJ4	GPIOPJ4	BOARD_ID0	
PJ5	GPIOPJ5	BOARD_ID1	
PJ6	GPIOPJ6	BOARD_ID2	
PJ7	GPIOPJ7	BOARD_ID3	
PK0	U4RX	FAST_PROCHOT_N	
PK1	U4TX	BMC_JTAG_SEL	PUSH PULL
PK2	GPIOPK2	FM_PWRBTN_OUT_N	OPEN DRAIN
PK3	GPIOPK3	FM_CPU_THERMTRIP_LVT3_N	
PK4	I2C3SCL	SMB_PCIE_EXP1_R_SCL	OPEN DRAIN
PK5	I2C3SDA	SMB_PCIE_EXP1_R_SDA	OPEN DRAIN
PK6	I2C4SCL	SMB_PCIE_RISER_R_SCL	OPEN DRAIN
PK7	I2C4SDA	SMB_PCIE_RISER_R_SDA	OPEN DRAIN
PL0	LPC0FRAME#	LPC_LFRAME_N_ESPI_CS0_BMC_N	
PL1	LPC0CLK	CLK_24M_66M_LPC0_ESPI_BMC	
PL2	LPC0CLKRUN#	FM_LPC_CLKRUN_R_N	
PL3	LPC0SERIRQ	IRQ_LPC_SERIRQ_ESPI_CS1_R_N	
PL4	LPC0SCI#	FM_CPU_MSMI_CATERR_LVT3_N	
PL5	LPC0PD#	IRQ_BMC_PCH_NMI_R	
PL6	USB0DP	USB2_BIC_DP	
PL7	USB0DM	USB2_BIC_DN	

PM0	LPC0AD3	LPC_LAD3_ESPI_IO0_R	
PM1	LPC0AD2	LPC_LAD3_ESPI_IO1_R	
PM2	LPC0AD1	LPC_LAD3_ESPI_IO2_R	
PM3	LPC0AD0	LPC_LAD3_ESPI_IO3_R	
PM4	RMII0-EN0RREF_CLK	FM_BIC_RST_RTCRST	PUSH PULL
PM5	SGPMI	SGPIO_BMC_DIN	
PM6	EN0CRS	PWRGD_CPU_LVC3_R	
PM7	EN0COL	PWRGD_SYS_PWROK	
PN0	U1RTS		
PN1	U1CTS	HSC_MUX_SWITCH	PUSH PULL
PN2	U1DCD	FM_FORCE_ADR_N	PUSH PULL
PN3	U1DSR		
PN4	U1DTR		
PN5	U1RI		
PN6	EN0TXER	FM_THERMTRIP_DLY_TO_PCH	
PN7	GPIOPN7	FM_BMC_CPU_PWR_DEBUG_N	OPEN DRAIN
PP0	SGPMCLK	SGPIO_BMC_CLK_R	PUSH PULL
PP1	SGPMLD	SGPIO_BMC_LD_R_N	PUSH PULL
PP2	USB0NXT	IRQ_PVCCIO_CPU_VRHOT_LVC3_N	
PP3	RTCCLK		
PP4	GPIOPP4	IRQ_PVDDQ_ABC_VRHOT_LVT3_N	
PP5	I2C2SCL	SMB_BIC_CPLD_R_SCL	
PP6	I2C2SDA	SMB_BIC_CPLD_R_SDA	
PP7	GPIOPP7	IRQ_PVCCIN_CPU_VRHOT_LVC3_N	
PQ0	SSI3CLK	BIC_SPICLK_R	PUSH PULL
PQ1	SSI3FSS	BIC_SPICS0_N_R	PUSH PULL
PQ2	SSI3XDAT0	BIC_SPIMOSI_R	PUSH PULL
PQ3	SSI3XDAT1	BIC_SPIMISO_R	
PQ4	DIVSCLK	IRQ_SML1_PMBUS_ALERT_N	
PQ5	RMII0-EN0RXD0	HSC_SET_EN	PUSH PULL
PQ6	RMII0-EN0RXD1	FM_BMC_PREQ_N_NODE_R1	OPEN DRAIN
PQ7	GPIOPQ7	FM_MEM_THERM_EVENT_LVT3_N	
PR0	I2C1SCL	SMB_IPMB_STBY_LVC3_R_SCL	OPEN DRAIN
PR1	I2C1SDA	SMB_IPMB_STBY_LVC3_R_SDA	OPEN DRAIN
PR2	GPIOPR2	IRQ_PVDDQ_DEF_VRHOT_LVT3_N	
PR3	GPIOPR3	FM_CPU_ERR0_LVT3_N	
PR4	GPIOPR4	FM_CPU_ERR1_LVT3_N	
PR5	U1RX	UART_BIC_R_RXD	PUSH PULL
PR6	U1TX	UART_BIC_TXD	PUSH PULL
PR7	GPIOPR7	FM_CPU_ERR2_LVT3_N	
PS0	GPIOPS0	IRQ_SML0_ALERT_R_N	
PS1	LPC0RESET#	RST_PLTRST_BMC_N	
PS2	GPIOPS2	FM_SLPS3_R_N	
PS3	GPIOPS3	DBP_SYSPWROK_R	OPEN DRAIN
PS4	GPIOPS4	IRQ_UV_DETECT_N	
PS5	GPIOPS5	FM_UV_ADR_TRIGGER_EN	OPEN DRAIN
PS6	GPIOPS6	IRQ_SMI_ACTIVE_BMC_N	

PS7	GPIOPS7	RST_RSMRST_BMC_N	
PT0	GPIOPT0	BMC_HARTBEAT_LED_R	PUSH PULL
PT1	GPIOPT1	IRQ_BMC_PCH_SMI_LPC_R_N	OPEN DRAIN
PT2	GPIOPT2		
PT3	GPIOPT3	FM_BIOS_POST_CMPLT_BMC_N	

9.3.7 PCH GPIOs

The table below lists the GPIO assignments for PCH. Only few GPIOs of PCH are used.

Table 9-10: PCH GPIOs

GPIO	Function (1)	Signal Name	I/O/BI	Push-pull/Open drain
GPD0		FM_PCH_PWR_DEBUG_N	O	OPEN DRAIN
GPD1	ACPRESENT	PU_ACPRESENT	I	
GPD2	GBE_WAKE#	TP_PCH_LAN_WAKE_N		
GPD3	PWRBTN#	FM_PCH_PWRBTN_OUT_N	I	
GPD4	SLP_S3#	FM_SLPS3_N	O	PUSH PULL
GPD5	SLP_S4#	FM_SLPS4_FROM_PCH_N	O	PUSH PULL
GPD6	SLP_A#	TP_PCH_SLPA		
GPD7		TP_PCH_GPD7		
GPD8	SUSCLK	TP_PCH_SUSCLK		
GPD9		TP_PCH_GPD9		
GPD10	SLP_S5#	TP_PCH_SLPS5_N		
GPD11	GBEPHY	TP_PCH_GPD11		
GPP_A0	RCIN#	LPC_RCIN_N_ESPI_ALERT1_N	I	
GPP_A1	LAD0	LPC_LAD0_ESPI_IO0	BI	PUSH PULL
GPP_A2	LAD1	LPC_LAD1_ESPI_IO1	BI	PUSH PULL
GPP_A3	LAD2	LPC_LAD2_ESPI_IO2	BI	PUSH PULL
GPP_A4	LAD3	LPC_LAD3_ESPI_IO3	BI	PUSH PULL
GPP_A5	LFRAME#	LPC_LFRAME_N_ESPI_CS0_N	O	PUSH PULL
GPP_A6	SERIRQ	IRQ_LPC_SERIRQ_ESPI_CS1_N	O	OPEN DRAIN
GPP_A7	PIRQA#	PU_IRQ_LPC_PIRQA_N_ESPI_ALERT0_N	I	
GPP_A8	ClKRUN#	FM_LPC_CLKRUN_N	I	
GPP_A9	CLKOUT_LCP0	CLK_24M_66M_LPC0_ESPI	O	
GPP_A10	CLKOUT_LPC1	TP_PCH_GPP_A10		
GPP_A11	GPP_A11	FM_LPC_PME_N	I	
GPP_A12		IRQ_PCH_SCI_WHEA_N	I	
GPP_A13	SUSWARN#_SUS	PU_EUP_LOT6_N	I	
GPP_A14	ESPI_RESET#	TP_PCH_ESPI_RESET		
GPP_A15	SUSACK#	PU_SUSACK_N	I	
GPP_A16	CLKOUT_LPC2	TP_PCH_GPP_A16		
GPP_A17		TP_PCH_GPP_A17		
GPP_A18		FM_BIOS_ADV_FUNCTIONS	I	
GPP_A19		FM_ME_RCVR_N	I	
GPP_A20		TP_PCH_GPP_A20		
GPP_A21		TP_PCH_GPP_A21		
GPP_A22		TP_PCH_GPP_A22		
GPP_A23		TP_PCH_GPP_A23		

GPP_B0	CORE_VID0	TP_PCH_GPP_B0_CORE_VID0		
GPP_B1	CORE_VID1	TP_PCH_GPP_B0_CORE_VID1		
GPP_B2		PU_PCH_VRALERT_N	I	
GPP_B3	CPU_GP2	FM_QAT_ENABLE_N	I	
GPP_B4	CPU_GP3	FM_QAT_SEL	I	
GPP_B5	SRCCLKREQ0#	FM_PCH_INTERPOSER_SEL1	I	
GPP_B6	SRCCLKREQ1#	FM_PCH_INTERPOSER_SEL2	I	
GPP_B7	SRCCLKREQ2#	TP_PCH_GPP_B7		
GPP_B8	SRCCLKREQ3#	TP_PCH_GPP_B8		
GPP_B9	SRCCLKREQ4#	FM_BOARD_REV_ID2	I	
GPP_B10	SRCCLKREQ5#	FM_TPM_MOD_PRES_N	I	
GPP_B11		FM_PMBUS_ALERT_B_EN	O	PUSH PULL
GPP_B12	GLB_RST_WARN_N#	RST_GLB_RST_WARN_N	O	PUSH PULL
GPP_B13	PLTRST#	RST_PLTRST_FROM_PCH_N	O	PUSH PULL
GPP_B14	SPKR	FM_PCH_BIOS_RCVR_SPKR	I	
GPP_B15		FM_CPU_ERR0_LVT3_N	I	
GPP_B16		FM_CPU_ERR1_LVT3_N	I	
GPP_B17		FM_CPU_ERR2_LVT3_N	I	
GPP_B18		FM_NO_REBOOT	I	
GPP_B19		FM_BOARD_SKU_ID5	I	
GPP_B20		FM_BIOS_POST_CMPLT_N	O	OPEN DRAIN
GPP_B21		TP_PCH_GPP_B21		
GPP_B22		FM_PCH_BOOT_BIOS_DEVICE	I	
GPP_B23	SML1ALRT#	FM_PCH_BMC_THRMTRIP_EXI_STRAP_N	O	OPEN DRAIN
GPP_C0	SMBCLK	SMB_HOST_STBY_BMC_LVC3_R2_SCL	O	OPEN DRAIN
GPP_C1	SMBDATA	SMB_HOST_STBY_BMC_LVC3_R2_SDA	BI	OPEN DRAIN
GPP_C2	SMBALERT#	PU_PCH_TLS_ENABLE_STRAP	I	
GPP_C3	SML0CLK#	SMB_SMLINK0_STBY_LVC3_SCL	O	OPEN DRAIN
GPP_C4	SML0DATA	SMB_SMLINK0_STBY_LVC3_SDA	BI	OPEN DRAIN
GPP_C5	SML0ALERT#	IRQ_SML0_ALERT_N	O	OPEN DRAIN
GPP_C6	SML1CLK	SMB_PMBUS_SML1_STBY_LVC3_R2_SCL	O	OPEN DRAIN
GPP_C7	SML1DATA	SMB_PMBUS_SML1_STBY_LVC3_R2_SDA	BI	OPEN DRAIN
GPP_C8		FM_PASSWORD_CLEAR_N	I	
GPP_C9		FM_MFG_MODE	I	
GPP_C10		TP_PCH_GPP_C10		
GPP_C11		TP_PCH_GPP_C11		
GPP_C12		FM_BOARD_REV_ID0	I	
GPP_C13		FM_BOARD_REV_ID1	I	
GPP_C14		FM_BMC_PCH_SCI_LPC_N	I	
GPP_C15		TP_PCH_GPP_C15		
GPP_C16		TP_PCH_GPP_C16		
GPP_C17		TP_PCH_GPP_C17		
GPP_C18		TP_PCH_GPP_C18		
GPP_C19		TP_PCH_GPP_C19		
GPP_C20		FM_THROTTLE_N	O	OPEN DRAIN
GPP_C21		TP_PCH_GPP_C21		
GPP_C22		IRQ_BMC_PCH_SMI_LPC_N	I	

GPP_C23		FM_CPU_CATERR_DLY_LVT3_R_N	I	
GPP_D0	Serial Blink	IRQ_BMC_PCH_NMI	I	
GPP_D1	Serial Blink	TP_PCH_GPP_D1		
GPP_D2	Serial Blink	TP_PCH_GPP_D2		
GPP_D3	Serial Blink	TP_PCH_GPP_D3		
GPP_D4	Serial Blink	FM_PLD_PCH_DATA	O	OPEN DRAIN
GPP_D5	SSPO_SFR	TP_PCH_GPP_D5		
GPP_D6	SSPO_TXD	TP_PCH_GPP_D6		
GPP_D7	SSPO_RXD	TP_PCH_GPP_D7		
GPP_D8	SSPO_SCLK	FM_UPLINK_SEL	I	
GPP_D9	SSATA_DEVSLP3	TP_PCH_GPP_D9		
GPP_D10	SSATA_DEVSLP4	TP_PCH_GPP_D10		
GPP_D11	SSATA_DEVSLP5	TP_PCH_GPP_D11		
GPP_D12	SSATA_SDATAOUT1	TP_PCH_GPP_D12		
GPP_D13	SML0BCLK	TP_PCH_GPP_D13		
GPP_D14	SML0BDATA	TP_PCH_GPP_D14		
GPP_D15	SSATA_SDATAOUT0	TP_PCH_GPP_D15		
GPP_D16	SML0BALERT#	FM_ME_PFR_1	O	OPEN DRAIN
GPP_D17	DMIC_CLK1	FM_ME_PFR_2	O	OPEN DRAIN
GPP_D18	DMIC_DATA1	TP_PCH_GPP_D18		
GPP_D19	DMIC_CLK0	FM_PS_PWROK_DLY_SEL_R	O	OPEN DRAIN
GPP_D20	DMIC_DATA0	TP_PCH_GPP_D20		
GPP_D21	IE_UART_RX	TP_PCH_GPP_D21		
GPP_D22	IE_UART_TX	TP_PCH_GPP_D22		
GPP_D23		TP_PCH_GPP_D23		
GPP_E0	SATAXPCIE0	TP_PCH_GPP_E0		
GPP_E1	SATAXPCIE1	TP_PCH_GPP_E1		
GPP_E2	SATAXPCIE2	TP_PCH_GPP_E2		
GPP_E3	CPU_GP0	FM_ADR_TRIGGER_R_N	I	
GPP_E4	SATA_DEVSLP0	TP_PCH_GPP_E4		
GPP_E5	SATA_DEVSLP1	TP_PCH_GPP_E5		
GPP_E6	SATA_DEVSLP2	TP_PCH_GPP_E6		
GPP_E7	CPU_GP1	FM_ADR_SMI_GPIO_R_N	I	
GPP_E8	SATA_LED#	TP_PCH_GPP_E8		
GPP_E9	USB2_OC0#	FM_USB_OC_N	I	
GPP_E10	USB2_OC1#	TP_PCH_GPP_E10		
GPP_E11	USB2_OC2#	TP_PCH_GPP_E11		
GPP_E12	USB2_OC3#	TP_PCH_GPP_E12		
GPP_F0	SATAXPCIE3	TP_PCH_GPP_F0		
GPP_F1	SATAXPCIE4	TP_PCH_GPP_F1		
GPP_F2	SATAXPCIE5	TP_PCH_GPP_F2		
GPP_F3	SATAXPCIE6	TP_PCH_GPP_F3		
GPP_F4	SATAXPCIE7	FM_BIOS_USB_RECOVERY	I	
GPP_F5	SATA_DEVSLP3	IRQ_TPM_SPI_N	I	
GPP_F6	SATA_DEVSLP4	JTAG_PCH_PLD_TCK	O	PUSH PULL
GPP_F7	SATA_DEVSLP5	JTAG_PCH_PLD_TDI	O	PUSH PULL
GPP_F8	SATA_DEVSLP6	JTAG_PCH_PLD_TMS	O	PUSH PULL
GPP_F9	SATA_DEVSLP7	JTAG_PCH_PLD_TDO	I	

GPP_F10	SATA_SCLOCK	TP_PCH_GPP_F10		
GPP_F11	SATA_SLOAD	TP_PCH_GPP_F11		
GPP_F12	SATA_SDATAOUT1	TP_PCH_GPP_F12		
GPP_F13	SATA_SDATAOUT0	TP_PCH_GPP_F13		
GPP_F14	SSATA_LED#	TP_PCH_GPP_F14		
GPP_F15	USB2_OC4#	TP_PCH_GPP_F15		
GPP_F16	USB2_OC5#	TP_PCH_GPP_F16		
GPP_F17	USB2_OC6#	TP_PCH_GPP_F17		
GPP_F18	USB2_OC7#	TP_PCH_GPP_F18		
GPP_F19	LAN_SMBCLK	TP_PCH_GPP_F19		
GPP_F20	LAN_SMBDATA	TP_PCH_GPP_F20		
GPP_F21	LAN_SMBALRT#	TP_PCH_GPP_F21		
GPP_F22	SSATA_SCLOCK	TP_PCH_GPP_F22		
GPP_F23	SSATA_SLOAD	TP_PCH_GPP_F23		
GPP_G0	FANTACH0	TP_FAN_PCH_TACH0		
GPP_G1	FANTACH1	TP_FAN_PCH_TACH1		
GPP_G2	FANTACH2	TP_FAN_PCH_TACH2		
GPP_G3	FANTACH3	TP_FAN_PCH_TACH3		
GPP_G4	FANTACH4	TP_FAN_PCH_TACH4		
GPP_G5	FANTACH5	TP_FAN_PCH_TACH5		
GPP_G6	FANTACH6	TP_FAN_PCH_TACH6		
GPP_G7	FANTACH7	TP_FAN_PCH_TACH7		
GPP_G8	FANPWM0	TP_FAN_PCH_PWM0		
GPP_G9	FANPWM1	TP_FAN_PCH_PWM1		
GPP_G10	FANPWM2	TP_FAN_PCH_PWM2		
GPP_G11	FANPWM3	TP_FAN_PCH_PWM3		
GPP_G12		FM_BOARD_SKU_ID0	I	
GPP_G13		FM_BOARD_SKU_ID1	I	
GPP_G14		FM_BOARD_SKU_ID2	I	
GPP_G15		FM_BOARD_SKU_ID3	I	
GPP_G16		FM_BOARD_SKU_ID4	I	
GPP_G17	ADR_COMPLETE	FM_ADR_COMPLETE	O	PUSH PULL
GPP_G18	NMI#	IRQ_NMI_EVENT_N	O	OPEN DRAIN
GPP_G19	SMI#	IRQ_SMI_ACTIVE_N	O	OPEN DRAIN
GPP_G20	SSATA_DEVSLP0	IRQ_SML1_PMBUS_ALERT_N	I	
GPP_G21	SSATA_DEVSLP1	FM_BIOS_IMAGE_SWAP_N	I	
GPP_G22	SSATA_DEVSLP2	TP_PCH_GPP_G22		
GPP_G23	SSATAXPCIE0	TP_PCH_GPP_G23		
GPP_H0	SRCCLKREQ6#	TP_PCH_GPP_H0		
GPP_H1	SRCCLKREQ7#	FM_SWAP_OVERRIDE_N	I	
GPP_H2	SRCCLKREQ8#	TP_PCH_GPP_H2		
GPP_H3	SRCCLKREQ9#	TP_PCH_GPP_H3		
GPP_H4	SRCCLKREQ10#	TP_PCH_GPP_H4		
GPP_H5	SRCCLKREQ11#	TP_PCH_GPP_H5		
GPP_H6	SRCCLKREQ12#	TP_PCH_GPP_H6		
GPP_H7	SRCCLKREQ13#	TP_PCH_GPP_H7		
GPP_H8	SRCCLKREQ14#	TP_PCH_GPP_H8		
GPP_H9	SRCCLKREQ15#	TP_PCH_GPP_H9		

GPP_H10	SML2CLK	SMB_PMBUS_SML2_STBY_LVC3_SCL	O	OPEN DRAIN
GPP_H11	SML2DATA	SMB_PMBUS_SML2_STBY_LVC3_SDA	BI	OPEN DRAIN
GPP_H12	SML2ALERT#	FM_ESPI_FLASH_MODE	I	
GPP_H13	SML3CLK	TP_PCH_GPP_H13		
GPP_H14	SML3DATA	TP_PCH_GPP_H14		
GPP_H15	SML3ALERT#	PU_ADR_TIMER_HOLD_OFF_N	I	
GPP_H16	SML4CLK	TP_PCH_GPP_H16		
GPP_H17	SML4DATA	TP_PCH_GPP_H17		
GPP_H18	SML4ALERT#	FM_LT_KEY_DOWNGRADE_N	I	
GPP_H19	SSATAXPCIE1	FM_PCH_10GBE_PCI_DISABLE_N	I	
GPP_H20	SSATAXPCIE2	TP_PCH_GPP_H20		
GPP_H21	SSATAXPCIE3	FM_PCH_10GBE_LAN_DISABLE_N	I	
GPP_H22	SSATAXPCIE4	TP_PCH_GPP_H22		
GPP_H23	SSATAXPCIE5	FM_SSATA_PCIE_M2_SEL	I	
GPP_I0	LAN_TDO	TP_PCH_GPP_I0		
GPP_I1	LAN_TCK	TP_PCH_GPP_I1		
GPP_I2	LAN_TMS	TP_PCH_GPP_I2		
GPP_I3	LAN_TDI	TP_PCH_GPP_I3		
GPP_I4	RESET_IN#	TP_PCH_GPP_I4		
GPP_I5	RESET_OUT#	TP_PCH_GPP_I5		
GPP_I6	RESET_DONE	TP_PCH_GPP_I6		
GPP_I7	LAN_TRST_IN	TP_PCH_GPP_I7		
GPP_I8	PCI_DIS	FM_PCH_10GBE_PCI_DISABLE_N	I	
GPP_I9	LAN_DIS	FM_PCH_10GBE_LAN_DISABLE_N	I	
GPP_I10		TP_PCH_GPP_I10		
GPP_J0	LAN_LED_P0_0	TP_PCH_GPP_J0		
GPP_J1	LAN_LED_P0_1	TP_PCH_GPP_J1		
GPP_J2	LAN_LED_P1_0	TP_PCH_GPP_J2		
GPP_J3	LAN_LED_P1_1	TP_PCH_GPP_J3		
GPP_J4	LAN_LED_P2_0	TP_PCH_GPP_J4		
GPP_J5	LAN_LED_P2_1	TP_PCH_GPP_J5		
GPP_J6	LAN_LED_P3_0	TP_PCH_GPP_J6		
GPP_J7	LAN_LED_P3_1	TP_PCH_GPP_J7		
GPP_J8	LAN_I2C_SCL_MDC_P0	TP_PCH_GPP_J8		
GPP_J9	LAN_I2C_SDA_MDIO_P0	TP_PCH_GPP_J9		
GPP_J10	LAN_I2C_SCL_MDC_P1	TP_PCH_GPP_J10		
GPP_J11	LAN_I2C_SDA_MDIO_P1	TP_PCH_GPP_J11		
GPP_J12	LAN_I2C_SCL_MDC_P2	TP_PCH_GPP_J12		
GPP_J13	LAN_I2C_SDA_MDIO_P2	TP_PCH_GPP_J13		
GPP_J14	LAN_I2C_SCL_MDC_P3	TP_PCH_GPP_J14		
GPP_J15	LAN_I2C_SDA_MDIO_P3	TP_PCH_GPP_J15		
GPP_J16	LAN_SDP_P0_0	TP_PCH_GPP_J16		
GPP_J17	LAN_SDP_P0_1	TP_PCH_GPP_J17		
GPP_J18	LAN_SDP_P1_0	TP_PCH_GPP_J18		
GPP_J19	LAN_SDP_P1_1	TP_PCH_GPP_J19		

GPP_J20	LAN_SDP_P2_0	TP_PCH_GPP_J20		
GPP_J21	LAN_SDP_P2_1	TP_PCH_GPP_J21		
GPP_J22	LAN_SDP_P3_0	TP_PCH_GPP_J22		
GPP_J23	LAN_SDP_P3_1	TP_PCH_GPP_J23		
GPP_K0	LAN_NCSI_CLK_IN	TP_CLK_50M_RMII		
GPP_K1	LAN_NCSI_TXD0	TP_RMII_PCH_TXD0		
GPP_K2	LAN_NCSI_TXD1	TP_RMII_PCH_TXD1		
GPP_K3	LAN_NCSI_TX_EN	TP_RMII_PCH_TX_EN		
GPP_K4	LAN_NCSI_CRS_DV	TP_RMII_PCH_CRS_DV		
GPP_K5	LAN_NCSI_RXD0	TP_RMII_PCH_RXD0		
GPP_K6	LAN_NCSI_RXD1	TP_RMII_PCH_RXD1		
GPP_K7		FM_PCH_GBE_DEBUG_EN	I	
GPP_K8	LAN_NCSI_ARB_IN	PD_RMII_PCH_CONN_ARB_IN	I	
GPP_K9	LAN_NCSI_ARB_OUT	PU_RMII_PCH_CONN_ARB_OUT	I	
GPP_K10	PE_RST#	RST_PCIE_PCH_PERST_N	I	
GPP_L2	TESTCH0_D0	PTI_DATA0	O	
GPP_L3	TESTCH0_D1	PTI_DATA1	O	
GPP_L4	TESTCH0_D2	PTI_DATA2	O	
GPP_L5	TESTCH0_D3	PTI_DATA3	O	
GPP_L6	TESTCH0_D4	PTI_DATA4	O	
GPP_L7	TESTCH0_D5	PTI_DATA5	O	
GPP_L8	TESTCH0_D6	PTI_DATA6	O	
GPP_L9	TESTCH0_D7	PTI_DATA7	O	
GPP_L10	TESTCH0_CLK	PTI_CLK	O	
GPP_L11	TESTCH1_D0	TP_PCH_GPP_L11		
GPP_L12	TESTCH1_D1	TP_PCH_GPP_L12		
GPP_L13	TESTCH1_D2	TP_PCH_GPP_L13		
GPP_L14	TESTCH1_D3	TP_PCH_GPP_L14		
GPP_L15	TESTCH1_D4	TP_PCH_GPP_L15		
GPP_L16	TESTCH1_D5	TP_PCH_GPP_L16		
GPP_L17	TESTCH1_D6	TP_PCH_GPP_L17		
GPP_L18	TESTCH1_D7	TP_PCH_GPP_L18		
GPP_L19	TESTCH1_CLK	TP_PCH_GPP_L19		

10 Power

10.1 Input Voltage Level

The expected nominal input voltage delivered by the Yosemite V3's power delivery subsystem is 12.5VDC; however, has a varying range of 11.5V to 13.5V. The Delta Lake 1S server shall accept and operation normally with an input voltage range of 12.5V +/- 7%.

10.2 1S Server Power Budget

The total power budget of the Delta Lake 1S Server shall not exceed the Yosemite V3's platform power budget that is to be allocated to each server blade. The max power budget design specs are listed below:

Table 10-1: Power budget for delta lake configurations

Configuration	Maximum TDP design spec	Peak power spec (2ms)
Delta Lake Server board with storage drives	210W	285W
Delta Lake Server board with 1U Front Expansion	270W	400W
Delta Lake Server board with 1U Front Expansion and 2U Expansion	470W	720W

The delta lake board supports server throttle through prochot logic to ensure system workloads do not overstress the power distribution component and trigger fault conditions. To enable the above power budget the power throttle levels on the server will be set to levels specified in Table 5-1Table 10-2 and Table 10-3. The TDP throttle thresholds are programmable and all throttle thresholds are selectable by BMC between 1U and 2U configs.

To maximize transient power budget, the server implements 3 type of power limiting prochot triggers:

1. OC_Warning: Triggered by average power exceeding system TDP limits
2. HSC_Timer: Tied to Hot swap over current protection, but triggers when before the overcurrent (OCP) timer expires (2ms).
3. Fast prochot: Triggers when system instantaneous power (<20us time window) exceeds design limits, before severe OCP protection meant for short circuit protection can trigger.

Table 10-2: Power throttle thresholds for 1U config

Purpose	Throttle protection	Design value (A)	Min Time window	Calculation
Ensure protection of input cable, rack power budget and make sure no defective components run continuously	HSC generated OC_warning	35	>20ms	Current: min (120% of system TDP, rack power allocation, cable limits, thermal limits for system) Time: min (max averaging capability of HSC, TDP tau)
Ensure peak current due to soft short failure conditions or component failure trigger throttle before shutting down server	HSC generated HSC_TIMER	40	>2ms	min (120% of system TDP with CPU in turbo assuming components in Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)
Ensure Power virus conditions throttle system before shutting down server through fast OCP	Fast prochot	50	<20us	Min (80% of Severe OCP, 120%*Pmax on all system components)

Table 10-3: Power throttle thresholds for 2U config

Purpose	Throttle protection	Design value	Min Time window	Calculation
Ensure protection of input cable, rack power budget and make sure no defective components run continuously	HSC generated OC_warning	50	>20ms	Current: min (120% of system TDP, rack power allocation, cable limits, thermal limits for system) Time: min (max averaging capability of HSC, TDP tau)
Ensure peak current due to soft short failure conditions or component failure trigger throttle before shutting down server	HSC generated HSC_TIMER	70	>2ms	min (120% of system TDP with CPU in turbo assuming components in Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)
Ensure Power virus conditions throttle system before shutting down server through fast OCP	Fast prochot	85	<20us	Min (80% of Severe OCP, 120%*Pmax on all system components)

The Delta lake hot swap controller is used to protect the upstream power distribution components from catastrophic faults, by triggering system power off for overcurrent conditions. System workloads are not expected to trigger these faults, throttle signaling is expected to bring down system power before fault conditions are reached.

10-4: Fault thresholds for 1U configs

HW protection	Design value (A)	Time window	Calculation
OCP fault	40	>4ms	min (120% of system TDP with CPU in turbo assuming Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)
Sever OCP fault	60	100ns	Options fixed by HSC, select by checking voltage drop for SCP condition on neighboring blade
Sled OCP fault	250	>4ms	min (120% of system TDP with CPU in turbo assuming Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)

10-5: Fault thresholds for 2U config

HW protection	Design value (A)	Time window	Calculation
OCP fault	70	>4ms	min (110% of system TDP with CPU in turbo assuming Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)
Severe OCP fault	105	100ns	Options fixed by HSC, select by checking voltage drop for SCP condition on neighboring blade
Sled OCP fault	250	>4ms	min (120% of system TDP with CPU in turbo assuming Pl2=1.5x TDP, rack peak power, SOA permitted by hotswap)

.

10.2.1 Expansion System Power Design Consideration

Each of the Front expansion connectors are expected to carry up to 6A and the Riser connectors for 2U expansion can carry up to 19.8A each for a total of 39.6A

As shown in section 5.1, the Delta Lake 1S Server shall support various expansion configurations and blade chassis sizes to accommodate various future use cases. However, the expansion subsystem must adhere to specific power design requirements:

1. The sustained current draw from the front expansion shall be at most 6.6A (1 connector) while the 2U expansion board must be limited to 19.8A each or 39.6A total (2 connectors)
2. Although the power budget may allow for high power commodities, feasibility is ultimately determined by whether the cooling solution implemented is adequate and does not introduce long term reliability degradation of the hardware

Note: Detailed design requirements for server expansion use cases is defined in the separate Delta Lake 1S Server Expansion Design Specification, please refer to the corresponding document for additional guidelines that may apply.

10.3 Capacitive Load

The capacitance on the input 12V power rail for Delta Lake 1S Server shall be optimized to meet the system's power supply requirements and does not cause instability. In addition, the Delta Lake 1S Server shall be designed such that it adheres to the maximum capacitive load allowable as defined in the Yosemite V3 Platform Design Specification document.

10.4 Hot Swap Controller Circuit

As described in section 7, the Delta Lake 1S Server shall implement a dedicated HSC which is expected to support the following:

1. In-rush current control when motherboard is inserted and powered up
2. MOSFETs must be kept within safe operating area during all operational conditions such as power on/off and fault conditions
3. Signals that indicate power status, alerts, interrupts are expected to allow rapid response upon impeding fault conditions and/or warnings

4. Current limit protection for over current and short circuit whereby overcurrent threshold should be configured to 40A without 2U expansion (70A with 2U expansion)
5. Undervoltage and overvoltage protection shall be configured to 10.2 V and 14 V respectively
6. Default HSC response for fault conditions shall be latch off with retry as stuff option
7. PMBus interface that supports the following features:
    a. Report voltage, current, and power (VIP) telemetry with accuracy of +/- 2.0% or better when operating above 10% of the maximum range
    b. Status registers that allow the definition of upper and lower critical thresholds for VIP of which are logged upon being triggered
8. Implements a fast (<20us) overcurrent monitoring scheme that generates an alert based on a remotely selectable threshold that triggers system throttling (Fast PROCHOT#) either using the HSC itself or external circuits. The recommended threshold for Fast PROCHOT# shall be lower than the severe overcurrent limit such that there is no tolerance overlap.
    a. Special consideration shall be taken to ensure that CPU performance is not impacted during extended peak power excursions; thus, sophisticated filter may be necessary to avoid pre-emptive throttling during such peak power events.
9. Average current limit (measured for 20ms window) protection is implemented by configuring one of the HSC Alert GPIO pins to trigger based on the programmed Overcurrent (OC) warning threshold. This is also routed to the PROCHOT logic to throttle the system if OC warn threshold is exceeded.

The voltage drop on the HSC current-sense resistor should be less than or equal to 25mV at full loading.

The power reporting of the HSC must be better than 2%, from 25W to full loading at room temperature. Further optimizations to power telemetry accuracy shall be performed through firmware based on characterized results from multiple boards based on entire load range and operating temperature requirements.

10.5 Power Sequence and Standby Power

Since the Delta Lake 1S Server relies on a single input power source from the Yosemite V3, there is no power sequence requirement from the platform perspective. The PCH will also require certain standby power rails to allow for certain operations and functions. Design must avoid any leakage paths among different power subsystems (e.g. Yosemite V3 platform, 1S Delta Lake Server, and expansion cards).

The 3.3V standby rail is generated using the unused rail of the VCCIO VR. However, to power the controller generating this standby rail, an intermediate 3.3V LDO is used, and sourced by the 5V board standby rail.

The 3.3V is also used to power the M.2 drives (boot drive and data drive) on the delta lake board. Both M.2 slots are power through load switches to enable managed power cycling.

10.6 VR Design and Efficiency

Delta lake utilizes universal VR controller and power stage footprints.

10-6: Power VR controller configuration

	Power rail	Controller	Power stage	
CPU	PVCCIN	5 mm x 5 mm	3/4 phase	5 mm x 6mm
	VSA	5 mm x 5 mm	1 phase	4mm x 5mm
	VCCIO	5 mm x 5 mm	1 phase	4mm x 5mm
	3.3V standby	5 mm x 5 mm	1 phase	4mm x 5mm
DIMM	Power rail	Controller	Power stage	
	PVDDQ_ABC	5 mm x 5 mm	1/2 phase	5 mm x 6mm
	PVDDQ_DEF	5 mm x 5 mm	1/2 phase	5 mm x 6mm

High efficiency Voltage Regulators (VR) shall be used on Delta Lake 1S Server with at least 91% efficiency over the 30% to 90% load range. If higher efficiency VRs are available at additional cost and/or design complexity, then the vendor is encouraged to present the tradeoffs prior to implementation.

10.7 Power Reading and Capping

The Delta Lake 1S Server is responsible for actively monitoring its own power consumption through the HSC. Like the last generation of 1S Server design Twin Lakes, the PCH is responsible for the calculation of a one second average based on the samples and report it upstream to the Bridge IC which is later polled by the BMC.

The Delta Lake 1S Server must have the ability to rapidly throttle itself to the lowest possible power state when the platform asserts the MASTER_THROTTLE signal, receives a request directly from the BMC, or power event reported by on-board power monitor.

A preferred power-capping implementation is to reduce the 1S server’s power consumption gradually with fine-grained power control by steps as small as 5 watts and to reach the control target power limit within 3 seconds. This process shall be smooth but fast, and the settled power value shall be within -3% of the target power limit set by the platform.

The server card will also have circuit to generate a POWER_FAIL_N signal to inform the 1S server that the 12V input power to the server is going to be cut off in certain amount of time (which is pre-defined by the platform). The 1S server can leverage this signal to develop mechanisms to protect critical data prior to a power outage.

10.8 Configuration support

The platform design supports up to 110W CPUs and multiple memory configurations. While the board layout supports the highest power config, the design implements 2 BOM configurations, one optimized for the lower power CPU + DIMMs and 2nd one supporting maximum power CPU.

10-7: CPU and DIMM config for delta lake

BOM config	CPU	DIMM config	DIMM VR
1	Upto 95W (3 phase VR)	6x32GB DIMM (DDR4 R-DIMM, Rank:2 DRAM width:x4, 1 DPC, 8Gb DRAM tech @3200 MT/s, 2x refresh rate, 3 DIMMs/VR)	1 phase
2	Upto 110W (4 phase VR)	6x128GB DIMM (DDR4 LR-DIMM, Rank:4 DRAM width:x4, 1DPC, 16Gb DRAM tech @3200 MT/s, 2x refresh rate,3 DIMMs/VR)	2 phases

11 Delta Lake 1S Server Management

The primary server management functions will be provided using a BMC on the platform. The BMC on the platform will use an I^{2}C bus as the management interface. This section identifies the required information that must be accessible from the BMC.

11.1 Overview

The diagram below gives an overview of Delta Lake 1S CPU server management scheme.

There are 2x classes of use cases.

Class 1 use case is with multi-host NIC shared by 1S server cards. In this use case, 1S CPU card interfaces baseboard to the left of the diagram below, and an optional 1U front expansion card to the right of the diagram below.

Base board has below major components:

- Multi-host NIC
- BMC
- CPLD for glue logic, power sequence, and PFR
- Front I/O

1U Front Expansion card has below major components:

- CPLD for expansion of I2C/UART/JTAG to M.2 slots
- Bridge IC for I2C and management
- USB Hub for expansion of USB to M.2 slots

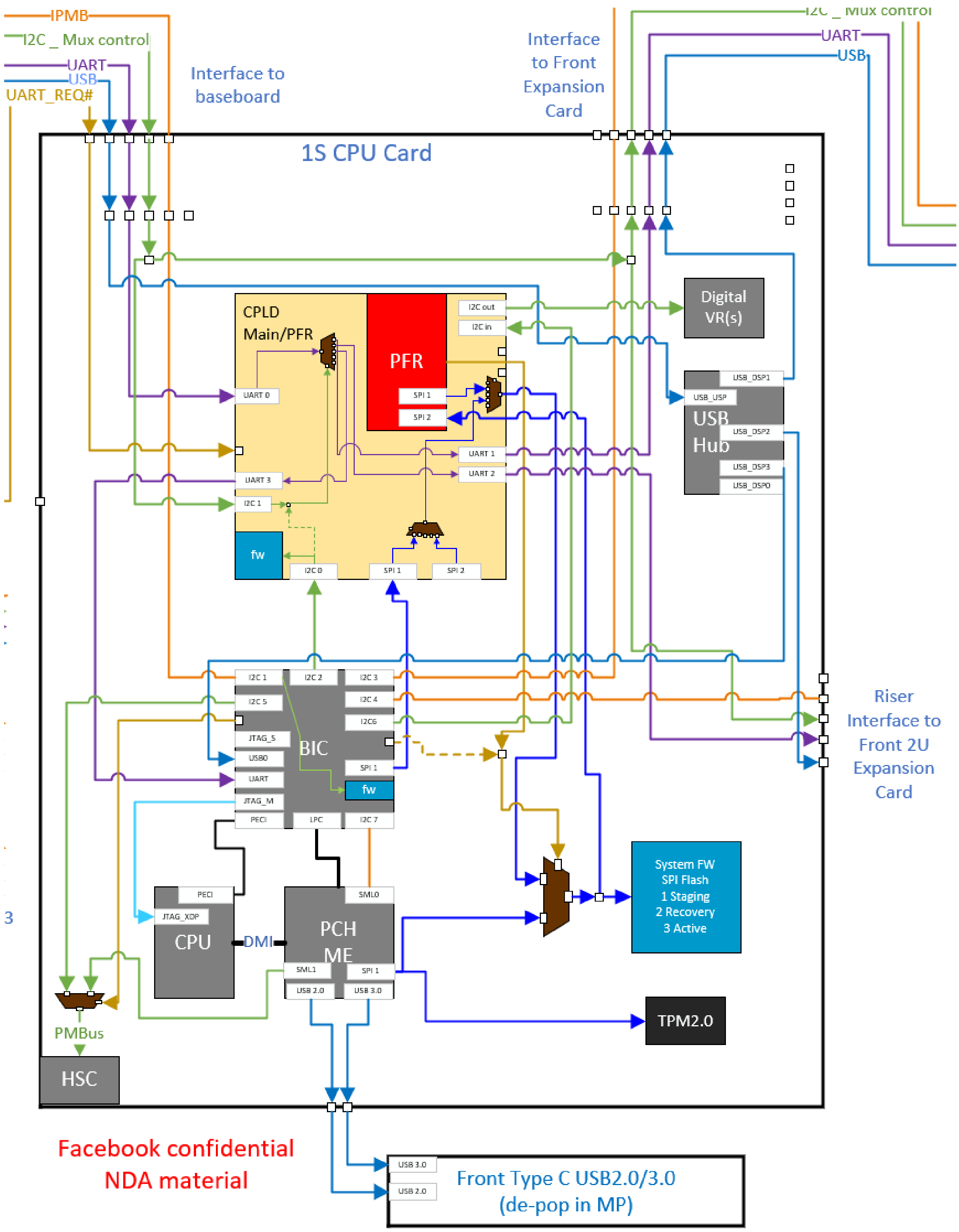

IPMB
I2C _ Mux control
UART
USB
UART_REQ#
Interface to baseboard
Interface to Front Expansion Card
I2C _ Mux control
UART
USB
1S CPU Card
CPLD Main/PFR
PFR
I2C out
I2C in
UART 0
SPI 1
SPI 2
UART 1
UART 2
UART 3
I2C 1
fw
I2C 0
SPI 1
SPI 2
Digital VR(s)
USB_DSP1
USB_USP
USB Hub
USB_DSP2
USB_DSP3
USB_DSP0
I2C 1
I2C 2
I2C 3
I2C 5
I2C 4
I2C6
JTAG_S
BIC
USB0
SPI 1
UART
fw
JTAG_M
PECI
LPC
I2C 7
Riser Interface to Front 2U Expansion Card
3
PECI
JTAG_XDP
CPU
DMI
SML0
PCH ME
SML1
SPI 1
USB 2.0
USB 3.0
System FW SPI Flash
1 Staging
2 Recovery
3 Active
TPM2.0
PMBus
HSC
Facebook confidential
NDA material
USB 3.0
USB 2.0
Front Type C USB2.0/3.0
(de-pop in MP)

Figure 11-1: Class 1 Server managerment

Class 2 use case has each 1S server card with single host NIC and its own BMC. In this use case, 1S CPU card interfaces baseboard to the left of the diagram below. On the right side, the 1S CPU card interface with 1) a NIC expansion card 2) a 2U front expansion card through a riser

Base board has below major components:

- CPLD for glue logic, power sequence
- A bridge IC
- Front I/O

NIC expansion card has below major components:

- A single host NIC
- BMC
- CPLD (Glue logic, power sequence, and PFR)

2U Front Expansion card has below major components:

- CPLD for expansion of I2C/UART/JTAG to M.2 slots
- USB Hub for expansion of USB to M.2 slots
- Bridge IC for I2C and management

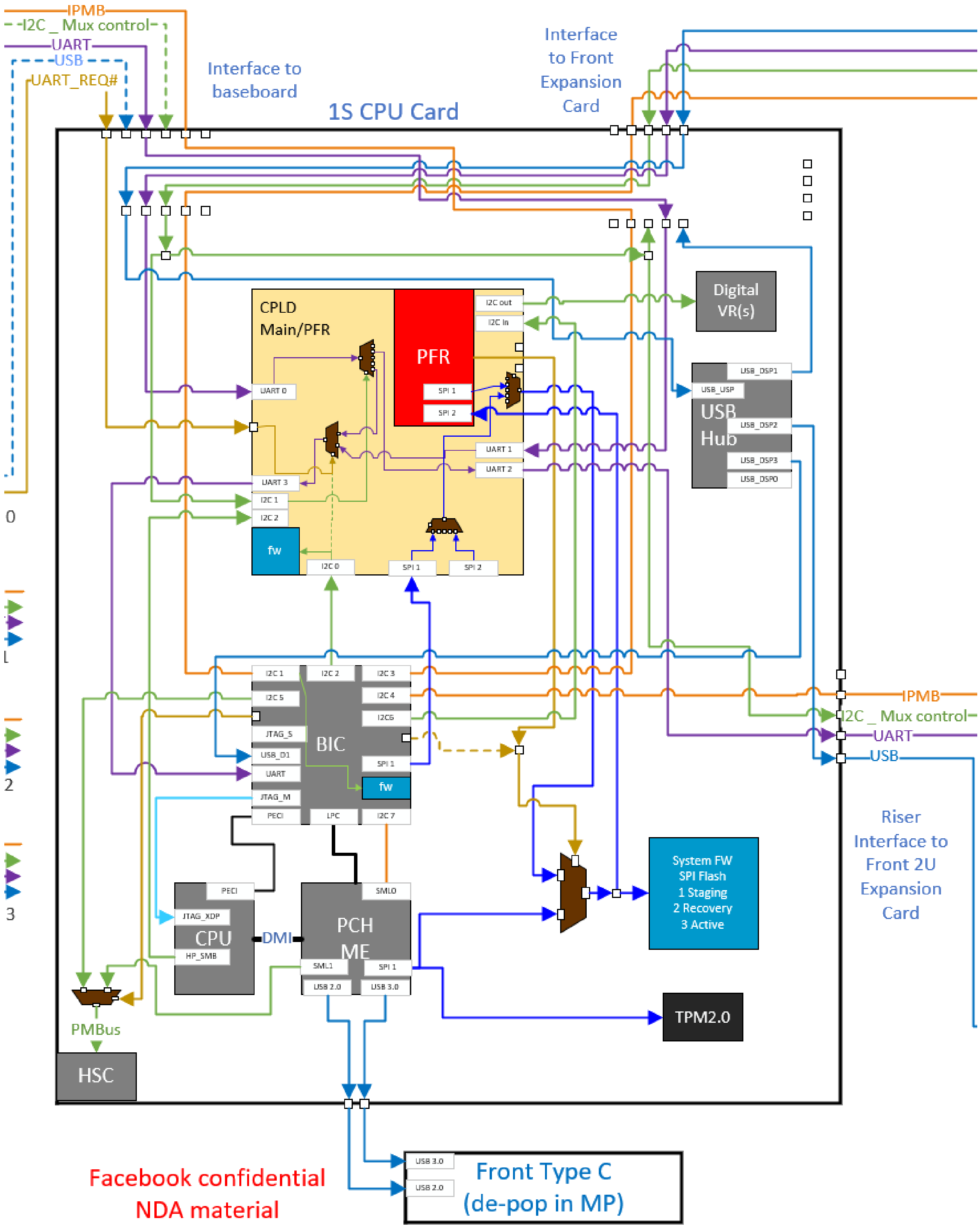


Figure 11-2: Class 2 server management

11.2 Bridge IC

The Delta Lake 1S server uses a Bridge IC, defined as the bridging device between the PCH and the BMC. The Bridge IC is on stand-by power so that it can be accessed by the BMC when the SoC is powered down.

On the platform side, the BMC and Bridge IC communicate with each other with IPMI messages over the I^{2}C bus. To enable prompt communication, this I^{2}C bus shall be a point-to-point link without any other devices on the same bus. It shall run in high-speed mode with 1MHz.

On the 1S server side, the Bridge IC has FRU EEPROM and thermal sensors on a local I^{2}C bus. The BMC can communicate with the Bridge IC to inquire the FRU and thermal data through the Intelligent Platform Management Bus (IPMB). The FRU EEPROM's data format is defined in Section 6.7. The thermal sensors are mainly used to measure the inlet and outlet temperatures of the 1S server for the platform thermal management's algorithm.

The Bridge IC has a dedicated I^{2}C bus to the PCH's Intel® Manageability Engine that supports IPMB. The Bridge IC now behaves as a transparent bridge to forward IPMI messages between the BMC and the PCH's Intel® Manageability Engine. With this transparent bridge, the BMC can directly work with the PCH's Intel® Manageability Engine to perform most of server management functions.

The Bridge IC also implements a system interface LPC to enable in-band server manageability. A KCS interface is implemented and it supports both standard SMM and SMS interfaces.

The Bridge IC monitors the Delta Lake 1S server's sensors, such as voltage sensors, power sensors, and digital sensors for critical GPIOs. The BMC can inquire about the 1S server's status by reading these sensors and taking actions via the Bridge IC.

It is recommended to use a versatile microcontroller as the Bridge IC. The microcontroller has a compact size, uses a low amount of power, and has adequate functions to support all required bridging functions. The Bridge IC on the Delta Lake 1S server is a Texas Instrument's Tiva microcontroller.

11.3 I^{2}C

The Delta Lake 1S server and BMC communicates using IPMI 2.0 commands transmitted over the I^{2}C connection through a Bridge IC on the 1S Server card. The I^{2}C bus address for the Bridge IC is configured as 0x40. The BMC on the platform is configured as 0x20.

The BIC on 1U Expansion card is configurated as 0x40.

The BIC on 2U Expansion card is configurated as 0x40.

Although all the BICs have the same address, it does not cause a conflict as the I^{2}C bus is point to point between the BICs and the BIC initiating the transaction does not respond to it's own request. This has been tested on Yosemite V3 platform.

11.4 IPMB Interface

The Bridge IC provides an IPMB interface for the BMC to access various IPMI resources on the Twin Lakes 1S Server. To meet this requirement, the Bridge IC shall implement various standard IPMI commands. It shall implement FRUID commands to identify the 1S server, System Event

Log (SEL) commands to store 1S Server specific event logs, and Sensor Data Repository (SDR) commands to identify various sensors described for the specific 1S server.

11.5 USB

Covered in section 9.3.3

11.6 SPI

Covered in section 9.3.4

11.7 JTAG

Covered in section 9.3.5

11.8 UART

The serial console of the Delta Lake server is used as the BIOS or OS serial console and will also be available as a Serial over Lan (SOL) connection via the BMC. BIC on Delta Lake talks to PCH over LPC and sends the serial data through CPLD to BMC. The BIOS menus must be fully accessible and text-based. Any hot keys that are required must be transmittable through a serial console session.

The BIOS should default to 57,600 bps/8N1.

11.9 Message Transfer

As the bridge between the server PCH and BMC on the platform, the Bridge IC provides ways to transfer messages between them via KCS interfaces.

For in-band management, the Bridge IC can forward the PCH's Keyboard Controller Style (KCS) request to the BMC and then send the received response back to the PCH.

When the BMC sends a request on the I^{2}C bus meant for the PCH's Intel® Manageability Engine, the Bridge IC shall forward the command on the I^{2}C bus and send the received response back to the BMC.

It is possible to implement an alternative SOL through the Bridge IC. When the alternative SOL feature is enabled, the serial data from the PCH's serial port shall be sent to the BMC via I^{2}C. When the BMC sends SOL data, it shall be emitted via the serial port.

11.10 Platform Discovery and Configuration

The Bridge IC provides a way for the BMC to discover platform capabilities such as electrical interface assignment. It also provides a way to discover and configure its own capabilities like enabling or disabling the SOL interface and/or POST code interface.

11.11 POST Code Access

During the power-on stage, the PCH usually sends out status/error information on the POST Code interface. The Bridge IC needs to provide a way for the BMC to access POST Code information. The Bridge IC shall keep the latest POST Code in a 230-byte buffer. Whenever the BMC is available, the Bridge shall send the POST Code as soon as it is received on the POST Code interface. Whenever the BMC is not available (such as the BMC being in a firmware update mode or during BMC boot-up) the Bridge IC shall add the latest POST Code at the top of the 230-byte buffer. The BMC shall be able to retrieve the POST Code buffer from the Bridge with the latest POST Code.

11.12 Firmware Update

The Bridge IC can update firmware of the programmable devices on the Delta Lakes 1S Server, such as BIOS, the PCH's Intel® Manageability Engine firmware, the CPLD image, various VR firmware and the Bridge IC's firmware.

Bridge IC provides a way for the BMC to access the version information and initiate update process of various firmware components on the Delta Lake 1S Server, as well as detect and retransmit corrupted firmware image packets during transit from the BMC to the Bridge IC.

Delta Lake will also implement Intel's Platform Firmware Resiliency which will require properly signed images in order to update the firmware on the CPLD as well as the BIOS. This will provide protection, detection and recovery in the case of an attempt to load improper or malicious firmware onto the system. This will provide checks at boot time, during firmware upgrade, and during runtime. This is described in more detail in Intel's documentation as well as Section 13.

11.13 GPIO Register (WIP table not updated)

The Bridge IC shall provide a GPIO interface to the BMC through a GPIO register block. In this way, the BMC can control the GPIO behind the Bridge IC (or this hardware abstraction layer) through accessing this register block.

The Bridge IC shall provide a way for BMC to configure GPIO pin direction, interrupt capability, and provide a way to get/set the current status of GPIO signals. It shall send an interrupt message to the BMC when the interrupt enabled GPIO signal changes its state. The GPIO register interface exposed by the Bridge IC shall provide signals that indicate various conditions as shown in Table 11-1.

Table 11-2: Bridge IC GPIO Register Table

	GPIO offset	GPIO Pin Function	Comments
	Byte 1 -bit [0]	Power Good – CPU Core	Indicates that the CPU's core power input is good
	Byte 1 - bit [1]	Power Good – PCH core	Indicates that the PCH's core power input is good
	Byte 1 - bit [2]	DDR Channel A/B Voltage regulator hot	Indicates that the DDR Channel A/B Voltage Regulator is hot
	Byte 1 - bit [3]	DDR Channel D/E Voltage regulator hot	Indicates that the DDR Channel D/E Voltage Regulator is hot

	Byte 1 - bit [4]	CPU VccIN Voltage regulator hot	Indicates that the CPU VccIN Voltage Regulator is hot
	Byte 1 - bit [5]	CPU Throttle	System firmware request CPU throttle
	Byte 1 - bit [6]	PCH Hot	Indicates PCH temperature is over setpoint
	Byte 1 - bit [7]	DIMM Hot	Indicates that one or more DIMM is hot
	Byte 2 - bit [0]	CPU thermal trip	Indicates that CPU experienced over temperature event and shut-down
	Byte 2 - bit [1]	PCH thermal trip	Indicates that PCH experienced over temperature event and shut-down
	Byte 2 - bit [2]	CPU FIVR fault	Indicates a CPU internal Voltage Regulator Error condition
	Byte 2 - bit [3]	CPU Catastrophic Error	Indicates that the CPU experienced catastrophic error
	Byte 2 - bit [4]	CPU Non-Recoverable Error	Indicates that the CPU experienced non-recoverable error
	Byte 2 - bit [5]	CPU Critical Error	Indicates that the CPU experienced critical error
	Byte 2 - bit [6]	CPU Non-Critical Error	Indicates that the CPU experienced non-critical error
	Byte 2 - bit [7]	Sleep S4 state	When low, indicates CPU has entered S4 state or lower
	Byte 3 - bit [0]	Non-maskable interrupt	Non-maskable interrupt
	Byte 3 - bit [1]	System management interrupt	System management interrupt
	Byte 3 - bit [2]	Platform Reset	Platform and PCIe Reset
	Byte 3 - bit [3]	Front Panel Reset Input	Initiate platform reset (front panel reset button being pressed)
	Byte 3 - bit [4]	Front Panel Reset Output	Host Reset Output from Bridge-IC
	Byte 3 - bit [5]	Bios Power-on Self Test (POST) complete	Bios Power-on Self Test (POST) complete

	Byte 3 - bit [6]	Sleep S3 state	When low, indicates CPU has entered S3 state or lower
	Byte 3 - bit [7]	Power Good – CPU VccIN	Indicates that the VCCIN Voltage Regulator power is good
	Byte 4 - bit [0]	Boot SPI selection	Select boot SPI0 or SPI1 as boot SPI
	Byte 4 - bit [1]	Ejector latch detection	Identify ejector fully closed
	Byte 4 - bit [2]	BMC Reset	Reset BMC from host
	Byte 4 - bit [3]	At-scale-debug (ASD) TCK selection	Select at-scale-debug (ASD) TCK drive CPU TCK or PCH TCK
	Byte 4 - bit [4]	BMC ready	Indicated BMC is ready
	Byte 4 - bit [5]	Host / Bridge-IC UART select	Select Host UART or Bridge-IC Debug UART
	Byte 4 - bit [6]	I2C MUX reset	Reset I2C MUX
	Byte 4 - bit [7]	At-scale-debug (ASD) PREQ	CPU probe mode request
	Byte 5 - bit [0]	At-scale-debug (ASD) JTAG TRST	JTAG Reset
	Byte 5 - bit [1]	System Throttle	Indicates system is currently throttling
	Byte 5 - bit [2]	At-scale-debug (ASD) PRDY	CPU probe mode ready
	Byte 5 - bit [3]	XDP Present	Indicates traditional ITP connected
	Byte 5 - bit [4]	ASD Present	Indicates at-scale-debug connected
	Byte 5 - bit [5]	CPU power debug	Use for debug CPU integrated VR
	Byte 5 - bit [6]	JTAG MUX selection	Select JTAG connection to ITP or ASD

11.14 IPMI commands

The Bridge IC must support the IPMI commands shown in Table 11-3.

Table 11-4: Bridge IC supported IPMI command Table

IPMI Command	Net Function	CMD#
Get Device ID	App	01h
Get Self Test Results	App	04h
Get System GUID	App	37h
Master Write-Read I^2C	App	52h
Get FRU Inventory Area Info	Storage	10h
Read FRU Inventory Data	Storage	11h
Write FRU Inventory Data	Storage	12h
Get SDR Repository Info	Storage	20h
Reserve SDR Repository	Storage	22h
Get SDR	Storage	23h
Get SEL Info	Storage	40h
Get SEL Allocation Info	Storage	41h
Reserve SEL	Storage	42h
Get SEL Entry	Storage	43h
Add SEL Entry	Storage	44h
Clear SEL	Storage	47h
Get Sensor Reading	Sensor/Event	2Dh
Get Sensor Reading	Sensor/Event	2Dh
Send request message to BMC	OEM (0x38)	01h
Send request message to Bridge-IC	OEM (0x38)	02h
Get all GPIO status	OEM (0x38)	03h

Set all GPIO status	OEM (0x38)	04h
Get GPIO configuration	OEM (0x38)	05h
Set GPIO configuration	OEM (0x38)	06h
Send interrupt to BMC	OEM (0x38)	07h
Send POST Code to BMC	OEM (0x38)	08h
Request POST Code data	OEM (0x38)	12h
Firmware Update	OEM (0x38)	09h
Firmware Verify	OEM (0x38)	0Ah
Get Firmware version	OEM (0x38)	0Bh
Enable Bridge IC update flag	OEM (0x38)	0Ch
Get NIC LED Frequency	OEM (0x38)	0Dh
Bridge IC Discovery	OEM (0x38)	0Eh
Platform Discovery	OEM (0x38)	0Fh
Set Bridge IC Configuration	OEM (0x38)	10h
Bridge IC Reset Cause	OEM (0x38)	11h
Bridge IC enter update mode	OEM (0x38)	13h
Set VR monitor Enable	OEM (0x38)	14h
Get VR monitor Enable	OEM (0x38)	15h
Reset BMC	OEM (0x38)	16h
Read BIOS image	OEM (0x38)	18h
Get Flash size	OEM (0x38)	19h
Set Pump Duty	OEM (0x38)	1Bh
Get Pump Duty	OEM (0x38)	1Ch

Set BIOS Chip Select	OEM (0x38)	1D
Get BIOS Chip Select	OEM (0x38)	1F
Set JTAG Tap State	OEM (0x38)	21
Shift JTAG Data	OEM (0x38)	22
Set System GUID	OEM (0x38)	EFh

Table 11-5 provides details of the IPMI Original Equipment Manufacturer (OEM) commands that are defined Table 11-6.

Table 11-7: Bridge IC supported IPMI command Table

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
01h	Send request message to BMC	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Request interface 01h: Intel® Manageability Engine 02h: SOL 03h: KCS SMS 04h: KCS SMM Byte 5:X – Request data Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – Request interface 01h: Intel® Manageability Engine 02h: SOL 03h: KCS Byte 6:X – Response data	This command is used for Bridge IC transfer request to BMC. For example: 1. Bridge IC gets "Get Device ID" command 0x06 0x01 from KCS. 2. Bridge IC will send this command to BMC as below. 0x38 0x01 0x03 0x06 0x01 3. BMC responds Get Device ID data.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
02h	Send request message to Bridge IC	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 –Receive interface 01h: Intel® Manageability Engine 02h: SOL Byte 5:X – Request data from BMC Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 –Receive interface 01h: Intel® Manageability Engine 02h: SOL Byte 6:X – Response data	This command is used for BMC send request to Bridge IC. For example: 1. When BMC want to send "Get Device ID" command to ME. It can use this command: 0x38 0x02 0x01 0x06 0x01 2. When Bridge IC receive this command, It will send "Get Device ID" command to Intel® Manageability Engine and get the response from Intel® Manageability Engine . 3. Bridge IC responds this command to BMC.
03h	Get all GPIO status	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:8 – Get all GPIO status 0b: Low 1b: High	This command used by BMC to get GPIO status from Bridge IC. Refer to Table 8 GPIO mapping table.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
04h	Set all GPIO status	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4:7 – GPIO enable mask, refer to GPIO mapping table 0b: Disable 1b: Enable Byte 8:11 – Set all GPIO status 0b: Low 1b: High Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	This command used by BMC to set GPIO status from Bridge IC. Refer to Table 8 GPIO Mapping Table.
05h	Get GPIO configuration	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4:7 – GPIO enable mask, refer to GPIO mapping table 0b: Disable 1b: Enable Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:X – GPIO configuration (one byte for one GPIO pin configuration) Bit[0] – Input/output pin 0b : Input pin 1b : Output pin Bit[1] – interrupt disable/enable 0b : Disable 1b : Enable Bit[2] – Edge trigger 0b : Edge trigger (default) Bit[3:4] – Trigger type 00b : Falling edge 01b : Rising edge 10b : Both 11b : Reserved	This command used by BMC to get GPIO configuration from Bridge IC. Refer to Table 8 GPIO Mapping Table.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
06h	Set GPIO configuration	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4:7 – GPIO enable mask, refer to GPIO mapping table 0b: Disable 1b: Enable Byte 8:X – GPIO configuration (one byte for one GPIO pin configuration) Bit[0] – Input/output pin 0b : Input pin 1b : Output pin Bit[1] – interrupt disable/enable 0b : Disable 1b : Enable Bit[2] – Edge trigger 0b : Edge trigger (default) Bit[3:4] – Trigger type 00b : Falling edge 01b : Rising edge 10b : Both 11b : Reserved Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	This command used by BMC to set GPIO configuration to Bridge IC. Refer to Table 8 GPIO Mapping Table.
07h	Send interrupt to BMC	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Interrupt GPIO number, refer to GPIO mapping table Byte 5 – Trigger type 00h: Falling edge 01h: Rising edge Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	This command used for Interrupt notification from Bridge IC sends to BMC. Refer to Table 8 GPIO Mapping Table.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
08h	Send POST Code to BMC	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Data length Byte 5:X – Port 80 data Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	Bridge IC support maximum 230 bytes to buffer BIOS POST Code when BMC is not ready. The POST Code data will be in FIFO manner i.e. with the first POST Code as the first byte. But in case BMC is ready, Bridge IC will send one POST Code to BMC at a time.
12h	Request POST Code data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:X – Port 80 data Byte 2:4 – IANA ID – 00A015h, LS byte first	BMC can get all POST Code data by this command. Bridge IC will buffer POST Code data for last boot. The POST Code data will be in LIFO manner i.e. with the latest POST code as the first byte. Bridge IC clear buffer when system power on. The maximum buffer data length is 230 bytes.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
09h	Firmware Update	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – update target 00h: BIOS 01h: CPLD - bit[7] = 1, last package for image data 02h: Bridge IC boot loader from OOB - bit[7] = 1, last package for image data 03h: Bridge IC boot loader from In-Band - bit[7] = 1, last package for image data 04h: VR - bit[7] = 1, last package for image data Byte 5:8 – Offset Byte 9:10 – Data length Byte11:X – Update image data Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) 80h –Write flash error 81h – Power status check fail 82h – Data length error 83h – Flash erase error Byte 2:4 – IANA ID – 00A015h, LS byte first	This command is used to update BIOS and CPLD Firmware from BMC.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
0Ah	Firmware Verify	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 –update target 00h: BIOS 01h: CPLD 02h: Bridge IC boot loader 03h: VR Byte 5:8 – Offset Byte 9:12 – Data length Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) 80h – Checksum error 82h – Data length error 84h – Read flash error Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:8 – Checksum	This command is used to verify BIOS and CPLD Firmware from BMC.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
0Bh	Get Firmware version	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 –update target 01h: CPLD 02h: Bridge IC 03h: Intel® Manageability Engine version 04h: Bridge IC Bootloader 05h: VCCIO VR 06h: VCCIN VR 07h: VCCSA VR 08h: DDR_AB VR 09h: DDR_DE VR 0Ah: VNNPCH VR 0Bh: 1V05PCH VR Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:X – CPLD version (Hexadecimal) – CPLD user code 4 bytes. Bridge IC version (Decimal) – 2 bytes length, ex: 1.03. Return data will be 0x01 0x03. Intel® Manageability Engine version (Decimal) – 5 bytes length, ex: Intel® version 03.0.0.010. Return data will be 0x03 0x00 0x00 0x01 0x00. VR version (Hexadecimal) – VR user data, 4 bytes length. Ex: 3d 01 11 00, user data 0 : 0x3d01, user data 1 : 0x1100 Bridge IC bootloader version (Decimal) – 2 bytes length, version 1.08Return data will be 0x01 0x08	
0Ch	Enable Bridge IC update flag	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Enable update interface flag 00h: UART 01h: I2C 02h: LPC Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	This command is used to enable Bridge IC update flag from BMC.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
0Dh	Get NIC LED frequency	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – LED frequency 00h: No blinking 01h: Solid on 02h: Slow flashing 03h: Fast flashing	
0Eh	Get Bridge IC configuration	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – Bit[0] – SOL interface 0b : Disable 1b : Enable Bit[1] – Port 80 0b : Disable, Bridge IC will not send post code to BMC 1b : Enable	

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
0Fh	Platform discovery	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – 0x03: A13/A14 PCIe0 RefClk Byte 6 – 0x02: A17/A18 PCIe0 Lane 0, Gen3 Byte 7 – 0x02: A21/A22 PCIe0 Lane 1, Gen3 Byte 8 – 0x02: A25/A26 PCIe0 Lane 2, Gen3 Byte 9 – 0x02: A29/A30 PCIe0 Lane 3, Gen3 Byte 10 – 0x05: A33/A34 SATA0, Gen3 Byte 11 – 0x03: A37/A38 PCIe2 RefClk Byte 12 – 0x02: A49/A50 PCIe1 Lane 0 Gen3 Byte 13 – 0x02: A53/A54 PCIe1 Lane 1 Gen3 Byte 14 – 0x02: A57/A58 PCIe1 Lane 2 Gen3 Byte 15 – 0x02: A61/A62 PCIe1 Lane 3 Gen3 Byte 16 – 0x02: A65/A66 PCIe2 Lane 0 Gen3 Byte 17 – 0x02: A69/A70 PCIe2 Lane 1 Gen3 Byte 18 – 0x02: A73/A74 PCIe2 Lane 2 Gen3 Byte 19 – 0x02: A77/A78 PCIe2 Lane 3 Gen3 Byte 20 – 0x02: B15/B16 PCIe0 Lane 0, Gen3 Byte 21 – 0x02: B19/B20 PCIe0 Lane 1, Gen3 Byte 22 – 0x02: B23/B24 PCIe0 Lane 2, Gen3 Byte 23 – 0x02: B27/B28 PCIe0 Lane 3, Gen3 Byte 24 – 0x05: B31/B32 SATA0, Gen3 Byte 25 – 0x03: B35/B36 PCIe1 Ref Clk Byte 26 – 0x02: B51/B52 PCIe1 Lane 0 Gen3 Byte 27 – 0x02: B55/B56 PCIe1 Lane 1 Gen3 Byte 28 – 0x02: B59/B60 PCIe1 Lane 2 Gen3 Byte 29 – 0x02: B63/B64 PCIe1 Lane 3 Gen3 Byte 30 – 0x02: B67/B68 PCIe2 Lane 0 Gen3 Byte 31 – 0x02: B71/B72 PCIe2 Lane 1 Gen3 Byte 32 – 0x02: B75/B76 PCIe2 Lane 2 Gen3 Byte 33 – 0x02: B79/B80 PCIe2 Lane 3 Gen3	For scalability needs, we list all configurations in this command response. If we store that information in BMC FW and check by FRU ID, we will need to modify BMC FW whenever there is a new card. We recommend storing the info on MB.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
10h	Set Bridge IC configuration	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Config Bridge IC Bit[0] – SOL interface 0b : Disable 1b : Enable Bit[1] – Port 80 0b : Disable, Bridge IC will not send post code to BMC 1b : Enable Bit[2] – KCS 0b : Disable, Bridge IC will not send KCS command to BMC 1b : Enable Bit[3] – IPMB message 0b : Disable, Bridge IC will not send IPMB message to BMC 1b : Enable Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	When BMC enter update mode, BMC can use this command to disable various communication to Bridge IC.
11h	Bridge IC reset cause	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Reset cause 0x00 : Cold reset by Firmware update 0x01 : Watchdog timeout Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	Bridge IC will send this command to notify BMC when Bridge IC is reset.
13h	Bridge IC enter update mode	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 2 – Firmware mode 0x01 : normal mode 0x0F : update mode Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	This command is used to notify BMC, Bridge IC enter update mode and normal mode before and after Firmware update.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
14h	Set VR Monitor Enable	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Enable/Disable VR Monitor 0x01 : normal mode 0x0F : update mode Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	
15h	Get VR Monitor Enable	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – VR Monitor 0x00: Disable 0x01: Enable	
16h	Reset BMC	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	
18h	Read BIOS image	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 –update target 00h: BIOS Byte 5:8 – Offset Byte 9 – Data length Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) 81h – Power status check fail Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:N – BIOS image data	

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
19h	Get Flash Size	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – target 00h: BIOS Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:8 – flash size	
1Bh	Set Pump Duty	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Duty Value, range from 0 to 100 Byte 5 – Pump number, in this project it's set to 0 Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	
1Ch	Get Pump Duty	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Pump number, in this project it's set to 0 Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – Pump Duty	
1Dh	Set BIOS Chip Select	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – BIOS chip select index 0x00: Select SPI 0 0x01: Select SPI 1 Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
1Fh	Get BIOS Chip Select	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Pump number, in this project it's set to 0 Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5 – BIOS Chip Select Index 0x00: select SPI 0 0x01: select SPI 1	
21h	Set JTAG Tap State	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – JTAG TMS Bit length Byte 5 – JTAG TMS Bit data Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first	
22h	Shift JTAG Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4:5 – write data bit length Byte 5:n – write data Byte n:n+1 – read data bit length Byte n+2 – last transaction index 0x00 – not the last transaction, keep the TMS tate 0x01 – last transaction, set tap state to Exit1 state Response: Byte 1 – Completion Code 00h – Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2) Byte 2:4 – IANA ID – 00A015h, LS byte first Byte 5:n – JTAG TDO response data	

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
EFh	Set System GUID	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4:19 – System GUID. See Picture 5, GUID format Response: Byte 1 – Completion Code 00h - Success (Remaining standard Completion Codes are shown in IPMI spec v2.0 table 5-2)	

11.15 Management interface

The PCH could have its own management controller. However, this controller must work together with the Bridge IC as well as the BMC to perform server management tasks. The management interface between the Bridge IC and the PCH consists of a supplier-agnostic interface that combines a simple register interface to abstract the card-specific details. This interface can be implemented in hardware, software, or a combination of the two.

11.16 Power Control

The BMC controls power on, off, and reset directly via the signals defined in the pin-out. If 12V to the card is lost and returns “AC Lost”, the BMC must be configurable to enable either an immediate power-on, delayed power-on, or the last power state prior to the event.

11.17 Thermal Alerts

The PCH provides a mechanism to provide thermal alerts and over temperature notifications. The BMC must be able to receive these alerts in a timely fashion to allow it to take action quickly. The I^{2}C alert signal must be used. In some cases, an over temperature condition may occur which forces the PCH to power-off immediately. This condition must be logged.

11.18 Sensors

The following list of analog and discrete sensors are provided and are reported by the Bridge IC to the BMC.

Analog sensors include:

- Outlet Temperature
- Inlet Temperature
- VR Temperature(s)
- VR Current(s)
- PCH Temperature
- CPU Thermal Margin
- DIMM Temperature(s)
- CPU Package Power
- CPU Tj$_{MAX}$
- Voltage Sensor(s)

- Current Sensor(s)
- Power(s)

Discrete sensors include:

- CPU Thermal Trip
- System Status
- CPU Fail
- System Boot Status
- PCH/DIMM Hot
- VR Hot

Event Only Sensors Include:

- Power Threshold Event
- POST Error
- Power Error
- CPU Throttle
- Machine Check Error
- PCIe Error
- Other IO Error
- Memory ECC Error

11.19 Event Log

The vendor should implement the BMC to support storing events/logs from each 1S server, baseboard, device carrier card, and mezzanine card.

11.20 Logged Errors

11.20.1 CPU Error

Both correctable ECC errors and uncorrectable ECC errors should be logged into the Event log. Error categories include Link and L3 Cache.

11.20.2 Memory Error

Both correctable ECC errors and uncorrectable ECC errors should be logged into the Event log. The Error log should indicate the location of the DIMM (if applicable), channel #, and slot #.

11.20.3 PCI-E Error

All errors, which have a status register, should be logged into the Event log, including root complex, endpoint devices, and any switch upstream/downstream ports if available. Link disable on errors should also be logged. The error classifications Fatal, Non-fatal, or Correctable follow the 1S server vendor's recommendation.

11.20.4 POST Error

All POST errors, which are detected by BIOS during POST, should be logged into the Event log.

11.20.5 Power Error

Two power errors should be logged. One is a 12.5V DC input power failure that causes all power rails on the baseboard to lose power, including standby power. The other is an unexpected system shutdown during system S0/S1 while the 12.5V DC input is still valid.

11.20.6 MEMHOT# and PROCHOT#

Memory hot errors and processor hot errors should be logged. The Error log should identify the error source as internal, coming from the processor or memory, or an external error coming from the voltage regulator.

11.21 LEDs and Buttons

As for the front panel IO for the server, the following are suggested

1. Power LED
2. Fault/LOC/Select LED
3. AC_cycle_on/off button

The Power and Fault/LOC LED behavior is as follows

Permitted States	Separate LEDs	
	PWR (Blue)	FAULT/LOC (Amber)
System AC Off/Service Action Allowed	⬛	⬛
System On/Status OK	🟦	⬛
System Off + Fault	⬛ / 🔷	🟨
System On + Locate	🟦	🔶
System Off + Locate	⬛ / 🔷	🔶
System On + Fault	🟦	🟨
System DC Off	🔷	⬛

As for the Select LED, its behavior is based on whether the server blade is selected for OCP Debug cards function. Select LED is Amber and is lit when the OCP Debug card serial port is connected to that Delta Lake through BMC.

AC_cycle button is meant to AC cycle the server blade independently by triggering the enable to its hot swap controller. This button has to be operational independent of SW state of the Delta Lake system. The enable to the hot swap controller is managed by BMC. With a powered on system, if the AC_cycle button is pressed for >4 seconds < 8 seconds, the button state is detected by BMC to trigger the AC cycling of Delta Lake. When AC_cycle button is pressed for >8 seconds, the BMC will trigger Delta Lake to turn AC off. When the system is powered off, if the AC_cycle button is pressed for <8 seconds, no action will be taken. If the AC_cycle button is pressed for >8 seconds, the server board will turn AC on.

12 System Firmware (BIOS)

The card supplier is responsible for supplying and customizing the BIOS for the CPU. The requirements are outlined in this section.

12.1 Specification compliance

- The system BIOS needs to comply with the following specification:
- UEFI Specification version 2.7
- ACPI Specification version 6.2.A
- SMBIOS Specification version 3.3

12.2 Configuration and Features

The BIOS is tuned to minimize card power consumption. It has the following features:

- Disables unused devices, including PCIe lanes, USB ports, SATA/SAS ports, etc.
- A BIOS setup menu
- The CPU settings can be tuned to achieve the optimal combination of performance and power consumption

12.3 BIOS Settings Tools

The card supplier shall provide a tool to make BIOS setting changes without requiring a BIOS re-flash. The BIOS settings update tool must also support success and failure codes so that updates can be easily scripted.

This tool has to be compatible in a Linux OS environment.

12.4 PXE Boot

The BIOS supports PXE boot and provides the ability to modify the boot sequence. When PXE booting, the card first attempts to boot from the first available Ethernet device.

The default boot device priority is:

1. USB device if available
2. Mezzanine card NIC IPv6
3. Mezzanine card NIC IPv4
4. PCIe M.2 or SATA M.2
5. SATA HDD

This process loops indefinitely and requires no user intervention.

12.5 iSCSI Boot

The BIOS shall be capable of iSCSI network boot.

12.6 Other Boot Options

The BIOS also supports booting from SATA/SAS and USB interfaces. The BIOS provides the capability to select different boot options.

12.7 BIOS Update

The BIOS can be updated from the OS under these scenarios:

- Scenario 1: Sample/Audit BIOS settings
- Return current BIOS settings, or
- Save/export BIOS settings in a human-readable form that can be restored/imported (as in Scenario 2)
- Scenario 2: Update BIOS with pre-configured set of BIOS settings
- Update/change multiple BIOS settings
- Reboot
- Scenario 3: BIOS/firmware update with a new revision
- Load new BIOS/firmware on machine and update, retain the current BIOS settings
- Reboot
- BIOS needs to support both vendor proprietary upgrade tool and EFI capsule loader to upgrade BIOS image.

Additionally, the update tools have the following capabilities:

- Update from the operating system.
- Can complete BIOS update or setup change with a single reboot (no PXE boot, no multiple reboots)
- No user interaction (such as prompts)
- BIOS updates and option changes do not take longer than fifteen minutes to complete

12.8 Remote BIOS Update

The BIOS can be updated remotely under these scenarios:

- Scenario 1: Sample/Audit BIOS settings
- Return current BIOS settings, or
- Save/export BIOS settings in a human-readable form that can be restored/imported (as in Scenario 2)
- Scenario 2: Update BIOS with a pre-configured set of BIOS settings
- Update/change multiple BIOS settings
- Reboot
- Scenario 3: BIOS/firmware update with a new revision
- Load new BIOS/firmware on machine and update, retain the current BIOS settings
- Reboot

Additionally, the update tools have the following capabilities:

- Update from the remote host over the LAN connection to BMC
- Can complete BIOS update or setup change with a single reboot (no PXE boot, no multiple reboots)
- No user interaction (like prompts)

- BIOS updates and option changes do not take longer than 20 minutes to complete
- Can be scripted and propagated to multiple machines

12.9 Logged Errors

The following list of errors is logged by the BIOS or Bridge IC. These errors must include the date, time, and location information so that failing components can be easily identified.

- CPU/Memory errors: Both correctable ECC and uncorrectable ECC errors are logged into the event log. Error categories include DRAM, Link, and others.
- PCIe* errors: Any errors that have a status register are logged into the event log, including root complex, endpoint device, and any switch upstream/downstream ports if available. Link disable on errors are also be logged. Fatal, non-fatal, or correctable error classification follows the chipset vendor's recommendation.
- POST errors: All POST errors detected by the BIOS during POST are logged into the event log.
- SATA or SAS errors: All correctable and uncorrectable errors are logged.
- System reboot events
- Sensor values exceeding warning or critical thresholds
- Power-startup sequencing failure events

The System Event Log (SEL) format and decoding scheme need to comply with the Facebook Unified SEL Format.

12.10 Error Thresholds

An error threshold setting must be enabled for both correctable and uncorrectable errors. Once the programmed threshold is reached, an event is triggered and logged.

- Memory Correctable ECC: The threshold default value is 1,000. When the threshold is reached, the BIOS logs the event and includes the physical DIMM location.

12.11 POST Codes

The BIOS outputs a set of Power-On Self-Test (POST) codes identifying the current initialization step and any errors encountered along the initialization. The output is provided on the serial console and errors are logged.

Errors in POST need to be sent to BMC in the form of SEL.

During the boot sequence, the BIOS shall initialize and test each DIMM module. If a module fails initialization or does not pass the BIOS test, the following POST Codes should indicate which DIMM has failed:

- Display sequence will be “00”, DIMM location, Major code and Minor code.
- The first hex character [00] is a starter. not standard for CPU number.
- DIMM location use points to the DIMM silk number of PCBA. Suppose a different PCBA silk number for each DIMM. The second hex character indicates the number of the DIMM module.
- The POST Code will also display both the error major code and minor code from the memory reference code.
- The display sequence will be “00”, DIMM location, Major code and Minor code with a one second delay for every code displayed.

- The BIOS shall repeat the display sequence indefinitely to allow time for a technician to service the system.

12.12 HOTPLUG support for PCIe devices

The BIOS need to be configured to allow PCIe hot-plug support. This is to ensure that any failure of the PCIe devices connected to the CPU will not bring down the system and allow continued operation in a reduced capacity manner. Logging of such event needs to be done.

- The HOTPLUG will be categorized in the two following scenarios:
  Software managed HOTPLUG: removal and addition of M.2 devices on a powered system with OS running. The software shall be staged to get ready for device removal.
    - The platform shall work collaboratively with the OS to ensure graceful removal and addition in this scenario. After the process is complete, the serviced device is supposed to come back online.
- Surprise hot removal: M.2 device is not behaving as expected due to errors either on the PCIe link or on the device itself alone. The expectation in this case is only on containing the error and ensuring system availability.

12.13 Latency for SMI error handling

It is expected that the error handling of SMI should have latency no more than 20mS. It is expected that this implementation is done using unified SEL for Error/Fault report.

12.14 Support of AC cycling of PCIe devices

In the event of device error for PCIe devices in the expansion systems (like accelerators for example), there exists a mechanism to power cycle these PCIe devices connected using the register in PCIe config space. This mechanism would need to be made working with devices connecting directly to the Root Complex (CPU) or the PCIe switch. The procedure would generally involve control of GPIOs upon toggling the register. This GPIO could be on the PCIe switch or a CPLD connecting to the reset, power and clock control to the device. It should be noted that for the case where a PCIe switch is not involved, this may involve the following:

- CPU to issue a I2C command thru its Hot Plug SMBUS controller to server board CPLD
- Server CPLD to relay this event to the server BIC either thru interrupt, or BIC polling.
- Server BIC to transfer this information to the expansion board BIC
- Expansion board BIC to sequence the control of power/reset to the relevant device with the expansion board CPLD as a logic gate.

There should be logging of such events when they happen. And there would also be “3-tries” mechanism to try to recover the erratic device in such a manner before the system gives up.

13 Platform Security

13.1 TPM (Trusted Platform Module)

CPU host complex (CPU+PCH) shall have access to a TPM 2.0 module on its SPI interface. This TPM will exist as a hardware TPM physically on the Delta Lake board. The TPM will be used for measured boot. It will hold image measurements until the system is fully booted and connected to the network. From there, these measurements will be used in order to attest the system during provisioning. The TPM will only be used to store measurement values and will not contain any keys used elsewhere in the system.

14 Environmental Requirements

The full system with the server card installed meets the following environmental requirements:

- Gaseous contamination: Severity Level G1 per ANSI/ISA 71.04-1985
- Storage relative humidity: 10% to 90% (non-condensing)
- Storage temperature range: -40°C to +70°C (long-term storage) *
- Transportation temperature range: -40°C to +70°C (short-term storage) *
- Operating altitude with no de-rating to 6000 ft

14.1 Vibration and Shock

The motherboard meets shock and vibration requirements according to the following IEC specifications: IEC78-2-(*) and IEC721-3-(*) Standard & Levels.

Table 14-1: Vibration and Shock Requirements

	Operating	Non-Operating
Vibration	0.5G rms, 5-500-5 Hz, Random vibe, 1 sweep, 20 minute along all three axes (+/-) 5-10 Hz – 0.5G 10-350 Hz – 1.2G 350-500 Hz– 0.5G	1.2G, 5 to 500 to 5 Hz per sweep, 1 sweep at 0.5 octave/minute, along three axes
Shock	2-6G, half sine, 11ms, total 6 shocks, along three axes (+/-)	12G, half sine, 11ms, total 6 shocks, along three axes (+/-)

15 Prescribed Materials

15.1 Disallowed Components

The following components are not used in the design of the motherboard:

- Components disallowed by the European Union's Restriction of Hazardous Substances Directive, **RoHS 2 Directive (2011/65/EU)**
- Trimmers and/or potentiometers
- Dip switches

15.2 Capacitors and Inductors

The following limitations apply to the use of capacitors:

- Only aluminum organic polymer capacitors made by high-quality manufacturers are used; they must be rated 105°C.
- All capacitors have a predicted life of at least 50,000 hours at 45°C inlet air temperature, under the worst conditions.
- Tantalum capacitors using manganese dioxide cathodes are forbidden.
- SMT ceramic capacitors with case size > 1206 are forbidden (size 1206 are still allowed when installed far from the PCB edge and with a correct orientation that minimizes the risk of cracking).
- X7R ceramic material for SMT capacitors should be used by default and at minimum X6S for portions of design subject to thermal hotspots such as CPU and/or DIMM cavities.
- COG or NP0 type should be used for tolerance sensitive portions of design
- Conditional usage of X5R ceramic material must be based on evaluation of worst-case thermal conditions and upon approval from Facebook

The following limitations apply to the use of inductors:

- Only SMT inductors may be used as the use of through-hole inductors is disallowed.

15.3 Component De-rating

For inductors, capacitors, and FETs, derating analysis is based on at least 20% derating.

16 Labels and Markings

The motherboard shall include the following labels on the component side of the motherboard. The labels shall not be placed in a way which may cause them to disrupt the functionality or the airflow path of the motherboard.

Description	Type	Barcode Required?
Safety Markings	Silk Screen	No
Vendor P/N, S/N, REV (Revision would increment for any approved changes)	Adhesive label	Yes
Vendor Logo, Name & Country of Origin	Silk Screen	No
PCB Vendor Logo, Name	Silk Screen	No
Date Code (Industry Standard: WEEK / YEAR)	Adhesive label	Yes
RoHS compliance	Silk Screen	No
WEEE Symbol. The motherboard will have the crossed out wheeled bin symbol to indicate that it will be taken back by the Manufacturer at the end of its useful life. This is defined in the European Union Directive 2002/96/EC of January 27, 2003 on Waste Electrical and Electronic Equipment (WEEE) and any subsequent amendments.	Silk Screen	No
CE Marking	Silkscreen	No
UL Marking	Silkscreen	No

17 Revision History

Author	Description	Revision	Date
Jia Ning	- Initial draft after change to CPX-6	0v10	2019/4/4
Jia Ning	- Reorganize the chapters - Refresh baseboard to 1S server card interface - Define class 1/2 use cases	0v11	2019/4/21
Kiran Vemuri	- Added the connector pin details & GPIO details - More detail around low speed interfaces	0v14	2019/8/7
Todd Westhauser Pavan Shetty Kiran Vemuri	- Added Security Section - Updated Connector Pinouts - Updated Power sections - Updated block diagram	0v15	2020/4/12
Todd Westhauser	- Updated DIMM Configurations	0v16	2020/5/26
Todd Westhauser	- Primarily Phrasing updates based on Intel Feedback	0v18	2020/8/31
Todd Westhauser	- Incorporating ODM Feedback	0v19	2021/1/14
Todd Westhauser	- Final Feedback Incorporated	1v00	2021/1/20