North Dome 1S Server Design Specification

0v7

Authors:

Chenyu Xu, Mechanical Engineer, Facebook

Damien Chong, Hardware Engineer, Facebook

Haoran Wu, Power Engineer, Facebook

Patrick Williams, Software Engineer, Facebook

Yin Hang, Thermal Engineer, Facebook

1 Copyrights and Trademarks

AMD® is a trademark of Advance Micro Devices, Inc in the U.S. and/or other countries.

EPYC™ is a trademark of Advance Micro Devices, Inc 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

Contributions to this Specification are made under the terms and conditions set forth in Open Web Foundation Contributor License Agreement ("OWF CLA 1.0") ("Contribution License") by:

Facebook Inc.

You can review the signed copies of the applicable Contributor License(s) for this Specification on the OCP website at http://www.opencompute.org/products/specsanddesign

Usage of this Specification is governed by the terms and conditions set forth in **Open Web Foundation Final Specification Agreement ("OWFa 1.0") ("Specification License").**

You can review the applicable Specification License(s) executed by the above referenced contributors to this Specification on the OCP website at http://www.opencompute.org/participate/legal-documents/

NOTWITHSTANDING THE FOREGOING LICENSES, THIS SPECIFICATION IS PROVIDED BY OCP "AS IS" AND OCP EXPRESSLY DISCLAIMS ANY WARRANTIES (EXPRESS, IMPLIED, OR OTHERWISE), INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, OR TITLE, RELATED TO THE SPECIFICATION. NOTICE IS HEREBY GIVEN, THAT OTHER RIGHTS NOT GRANTED AS SET FORTH ABOVE, INCLUDING WITHOUT LIMITATION, RIGHTS OF THIRD PARTIES WHO DID NOT EXECUTE THE ABOVE LICENSES, MAY BE IMPLICATED BY THE IMPLEMENTATION OF OR COMPLIANCE WITH THIS SPECIFICATION. OCP IS NOT RESPONSIBLE FOR IDENTIFYING RIGHTS FOR WHICH A LICENSE MAY BE REQUIRED IN ORDER TO IMPLEMENT THIS SPECIFICATION. THE ENTIRE RISK AS TO IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION IS ASSUMED BY YOU. IN NO EVENT WILL OCP BE LIABLE TO YOU FOR ANY MONETARY DAMAGES WITH RESPECT TO ANY CLAIMS RELATED TO, OR ARISING OUT OF YOUR USE OF THIS SPECIFICATION, INCLUDING BUT NOT LIMITED TO ANY LIABILITY FOR LOST PROFITS OR ANY CONSEQUENTIAL, INCIDENTAL, INDIRECT, SPECIAL OR PUNITIVE DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND EVEN IF OCP HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3 OCP Tenets Compliance

3.1 Openness

The North Dome Design Specification exemplifies openness by providing a design for a server CPU card which can be used to support numerous use cases range from webservers to storage to accelerators. This server design can be used as a standalone device in the Yosemite V2 platform or can be accompanied with use case specific expansion cards to help customize for other solutions.

3.2 Efficiency

North Dome features AMD CPU when paired with efficient low power sku of AMD CPU such as from Milan 95W family, achieve significant performance and performance/watt gain compared to Twin Lakes which helps to drive lower number of servers needed to fulfill capacity. This saves cost and nature resources.

3.3 Impact

North Dome is another addition to the Yosemite V2 server card family which already has its sets of expansion capabilities. North Dome bring alternative CPU vendor option to a stable platform and demonstrate superior performance and performance/watt capabilities

4 Scope

This specification describes the design of the North Dome 1S server based on AMD SP3 socket family. The design is capable of AMD Rome Zen2 and Milan Zen3 CPU but only AMD Milan Zen3 CPU is rigorously tested for functionality.

5 Contents

6 Overview

This document describes a single socket server design based on the AMD® EPYC™ Rome SoC and forward compatible with Milan SoC, which is referred to hereinafter as North Dome 1S server.

The North Dome 1S server is designed to use AMD® EPYC™ Rome or Milan SoC utilizing the performance of AMD® Zen2™ / Zen3™ 7nm microarchitecture into a dense, low-power SoC.

Powered by AMD® EPYC™ SoC, integrating multi-core CPU, memory controller, and superior input/output (IO) device capabilities, the North Dome 1S server is great for single or multi node platforms where a solid balance of performance & IO density, as well as optimal performance per unit power are key criteria. However, platform designers need to provide adequate power and cooling capabilities to handle the SoC's power and thermal requirements.

Below figure illustrates the North Dome 1S server block diagram.

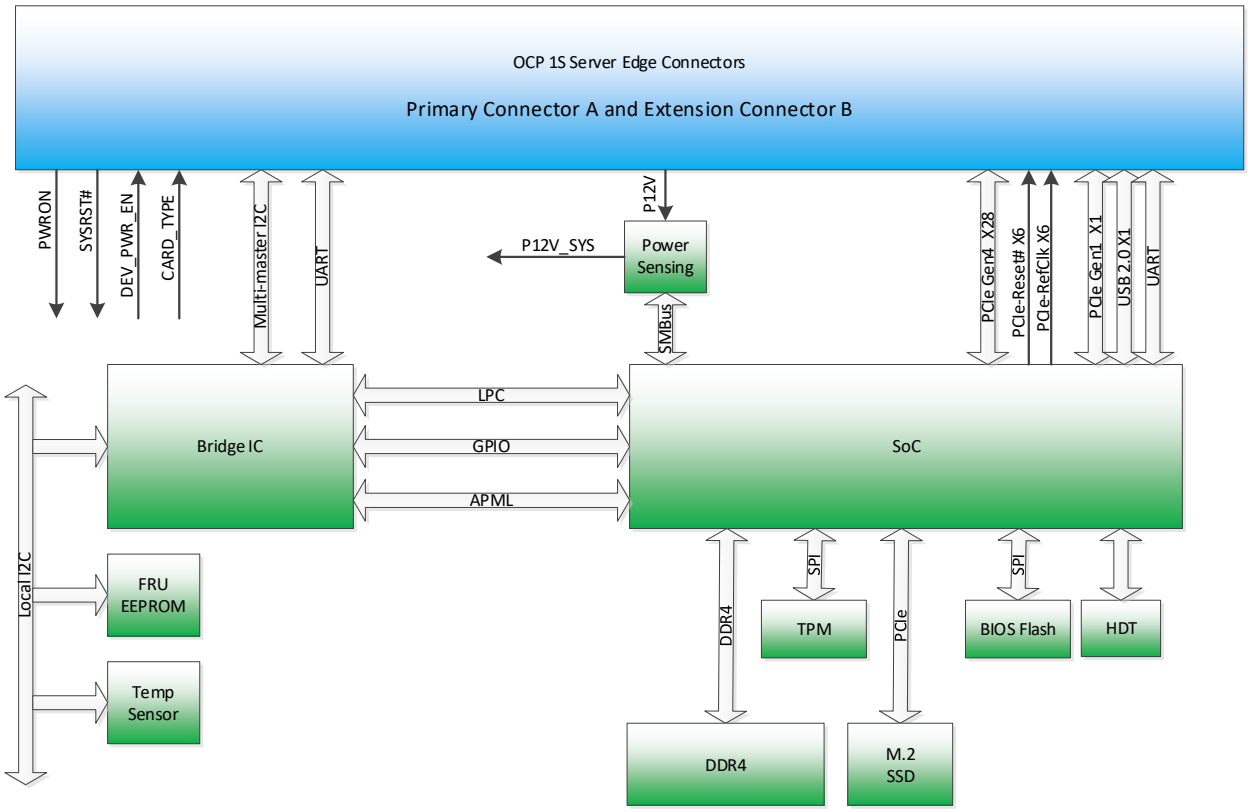

Figure 6-1: North Dome 1S Server Block Diagram

The North Dome 1S server implements primary and extension x16 PCIe edge connectors as a subset of pins defined in the 1S server specification.

The primary x16 PCIe edge connector supports:

- PCIe Gen3 ports
- A USB 2.0 port
- A Universal Asynchronous Receiver/Transmitter (UART)
- A multi-master I^2C bus for server management

The extension x16 PCIe edge connector supports:

- PCIe Gen3 ports

The North Dome 1S server supports two on-card Solid State Drives (SSDs), one in the 2280 and another in 22110 M.2 form factor.

The 2280 M.2 SSD is typically used as boot drive, it is on secondary side of the card. It has x4 PCIe connected to the SoC. The recommended minimum capacity of the boot drive is 256GB.

The other 22110 M.2 SSD drives are targeted for high performance use-cases, it is connected with x4 PCIe link to the SoC.

The North Dome 1S server only supports external network interface controller on the platform through its PCIe interface.

The North Dome 1S server receives 12.5V from the platform with a maximum current load of 7.7A from the primary edge connector and an additional 7.7A from the extension edge connector. The platform, however, defines and controls the maximum power used by the North Dome 1S server. The North Dome 1S server uses the discrete power sensor at the power input to measure the total card power consumption of the whole server card with +/-1% accuracy. The power data measured by this sensor can be used for power management purposes. When the power consumption has exceeded the power limit, the Bridge IC will generate a throttle signal to alert the SoC throttle to the lowest power state. The North Dome 1S server supports an Advanced Configuration Power Interface (ACPI)-compliant power button and reset signals from the platform.

A Bridge IC (Texas Instrument's Tiva microcontroller) is used as the management controller on the 1S server and the bridge between the BMC and the SoC. The Bridge IC manages the 1S server on behalf of the BMC on the platform and bridges the BMC and SoC. To maximize the communication bandwidth between the BMC and Bridge IC, a dedicated multi-master point-to-point I^2C bus shall be used.

North Dome 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 SoC for error reporting and other management purposes. A Low Pin Count (LPC) bus between the Bridge IC and SoC is connected to enable in-band communications. The BMC can access the North Dome 1S server's thermal sensors, FRU, SoC's GPIOs and SoC's management via the Bridge IC with standard IPMI commands.

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

7 Functional

7.1 CPU

North Dome 1S server will be built with an AMD SP3 socket compatible CPU. While the design is capable of AMD Zen2 Rome and AMD Zen3 Milan CPU, only AMD Zen3 Milan CPU with cTDP of 85W-105W has been rigorously tested with North Dome.

North Dome features a socketed CPU which can be field replaced. However, it would require tools for such replacements. For adopters who prefer quick and tool-less replacements, one way is to swap CPU back at integrator's factory.

7.2 Memory

The North Dome 1S server uses DDR4 memory bus, expected to run at 2667MT/s minimum, with design target to hit maximum speed of 3200MT/s.

6-channel DIMM connector and its associated channel assignment as follows.

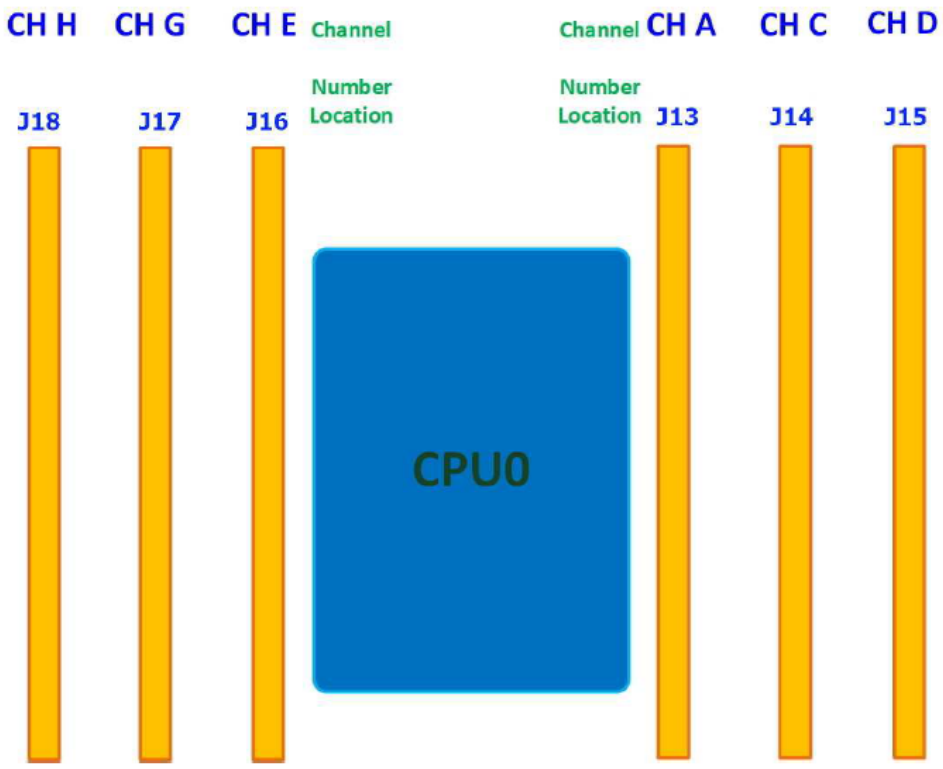

Figure 7-1: Memory channel naming convention

7.3 CPU debug interface

The North Dome 1S server shall support AMD HDT CPU debug header.

7.4 Storage SSD

The North Dome 1S server has two M.2 solid-state boot drive in 2280 and 22110 form factors respectively. This drive is connected to CPU through PCIe interface running NVMe protocol.

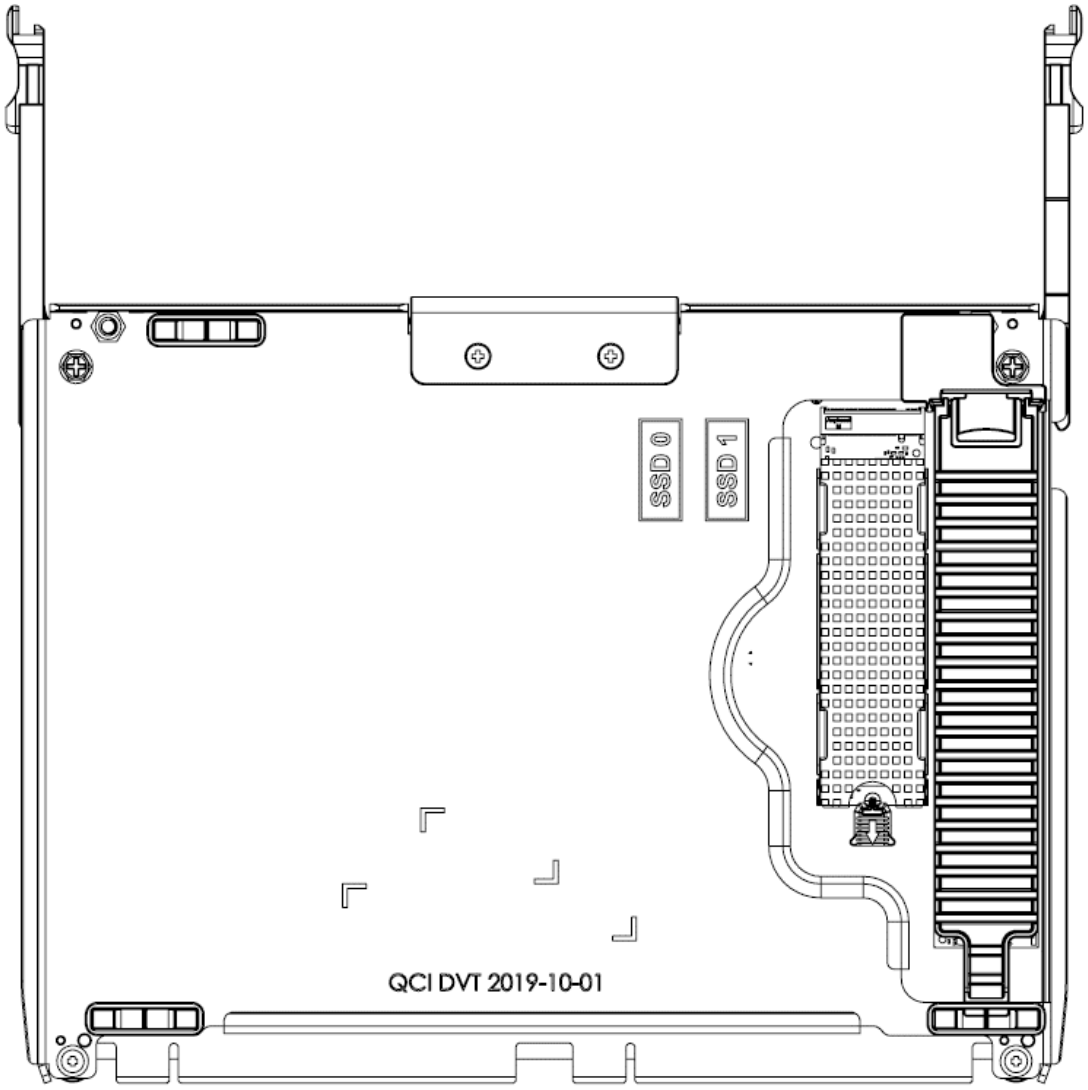

Figure 7-2: M.2 SSD on North Dome Motherboard

7.5 Expansion systems

The North Dome 1S server module has 28 PCIe Gen3 IO expansion connectivity to NIC and other peripherals devices thru two edge gold fingers.

When paired with Yosemite V2 baseboard, 4 PCIe Gen3 lanes being connected to NIC. While other 24 PCIe lanes can be connected to PCIe CEM expansion thru Crane Flat expansion card or additional M.2 SSDs thru Glacier Point V1 and V2 carrier.

7.6 FRU EEPROM

The North Dome 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)

8 Mechanical

8.1 Mechanical Outline

The North Dome 1S Server card uses a primary x16 PCIe edge connector and an additional extension x16 PCIe edge connector as the interfaces to the platform.

The overall dimensions of the general card is 210mmx160mm. See Figure 8-1: North Dome 1S Server Mechanical Drawings for the specification drawing.

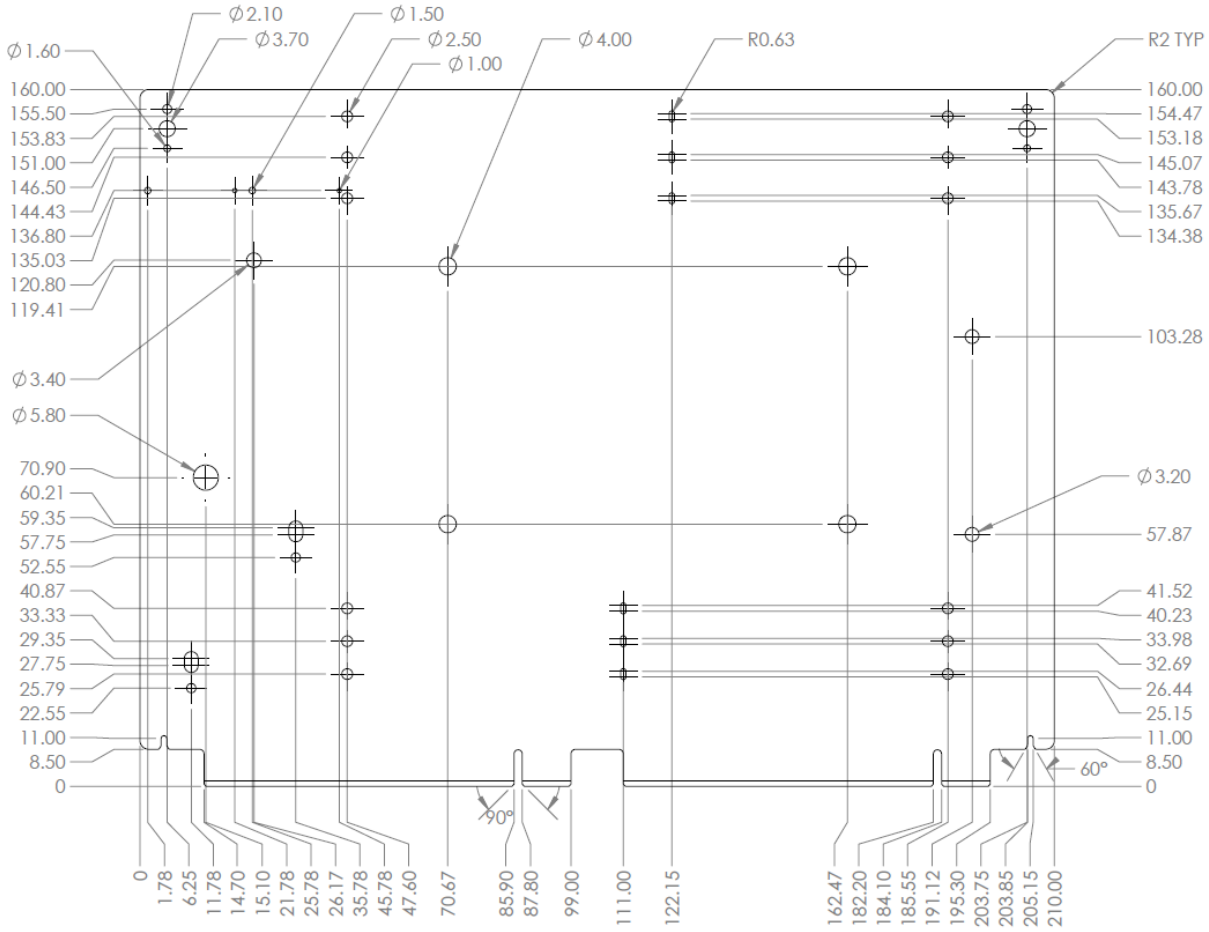

Figure 8-1: North Dome 1S Server Mechanical Drawings

8.2 Mechanical Details

The PCB is mounted to a sheet metal carrier which includes flanges on each perpendicular edge that engage the card guides in the chassis. There are no card guide keep-outs on the PCB edge since all available space is needed for traces and components.

The carrier assembly includes 2x ejectors which are used for card injection/ejection into the PCIe connectors. These ejectors have a push mechanism before being allowed to swing open, and they include finger access cutouts. One ejector claw engages a limit switch, which allows the BMC to detect if the ejector has been opened. The air duct rotates and latch on a fixed support wall.

A high-performance passive cooling solution chosen to cool the SoC effectively.

The thickness of the PCB shall be 1.57±10% mm to accommodate regular PCIe x16 connectors on platforms. Components of significant height will be placed on the Primary/A-side with maximum height limit of 45mm. Low profile components of height less than 2.67mm can be placed on the Secondary/B-side with exception of M.2s and HDT connector for debug. Boot drive M.2 connector shall be 5.8mm connector only. High performance M.2 has the option of 5.8mm connector or enhance serviceability with 6.7mm connector.

The next few figures show North Dome server card from different views.

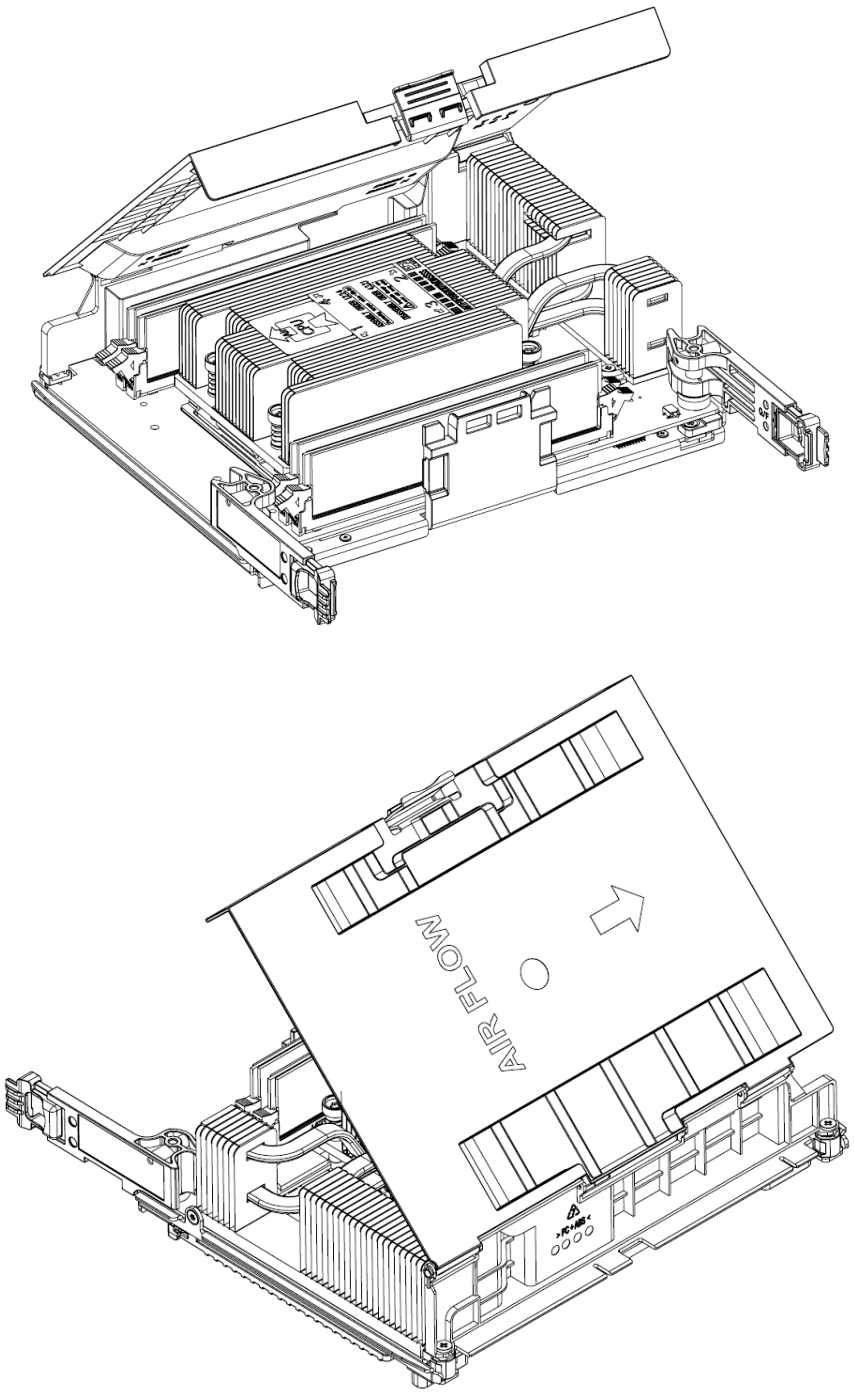

Figure 8-2: Isometric Views of North Dome 1S Server Assembly

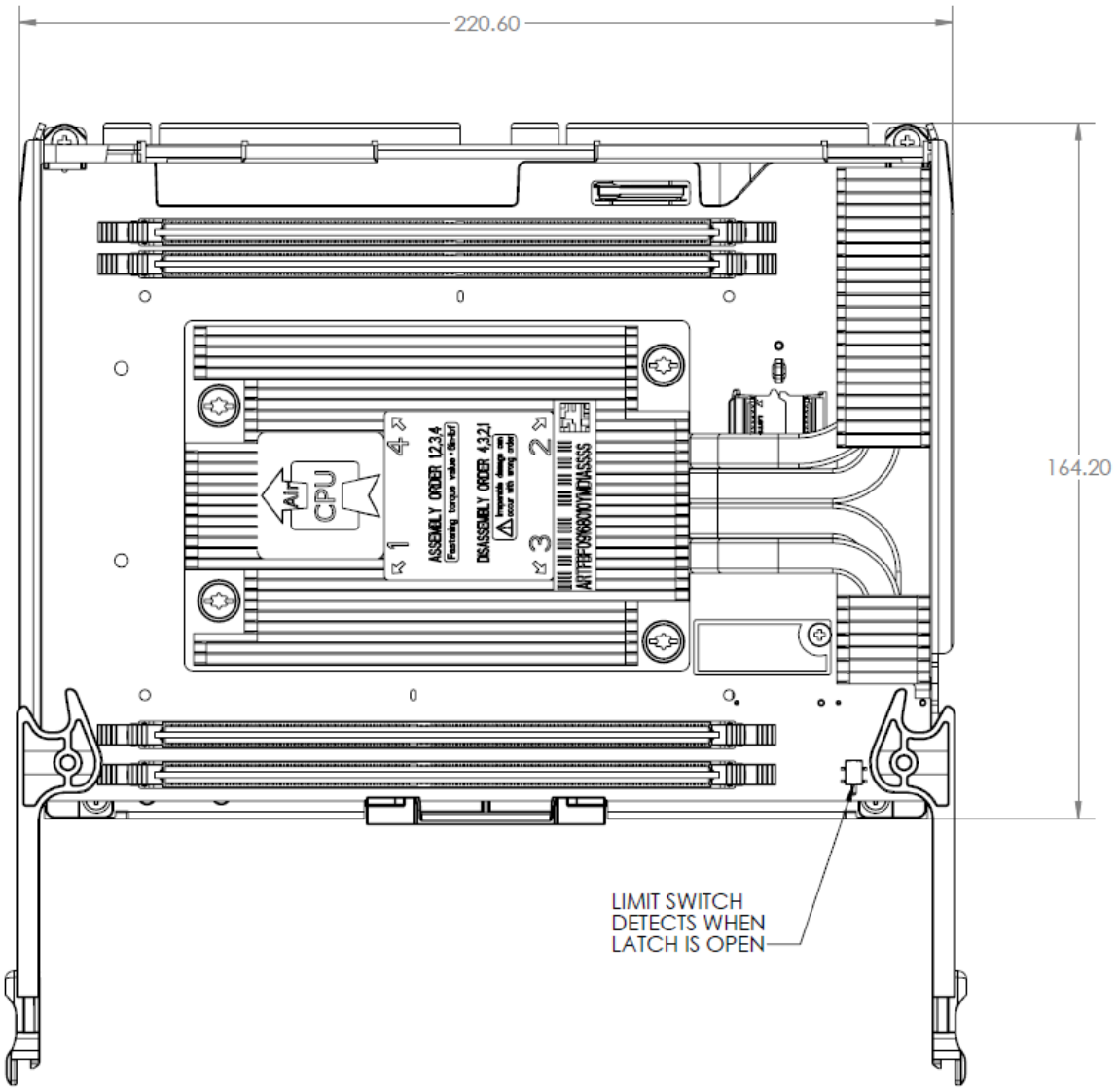

Figure 8-3: Topdown view of North Dome 1S Server Assembly

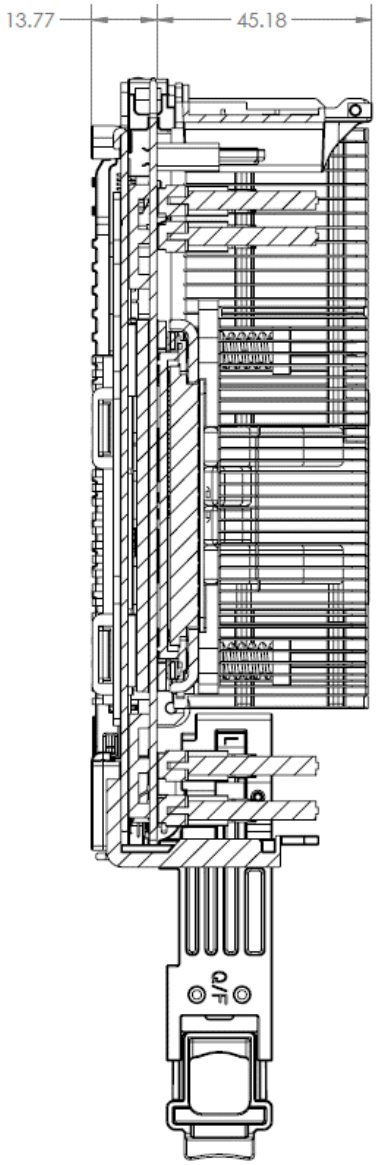

Figure 8-4: Side view of North Dome 1S Server Assembly

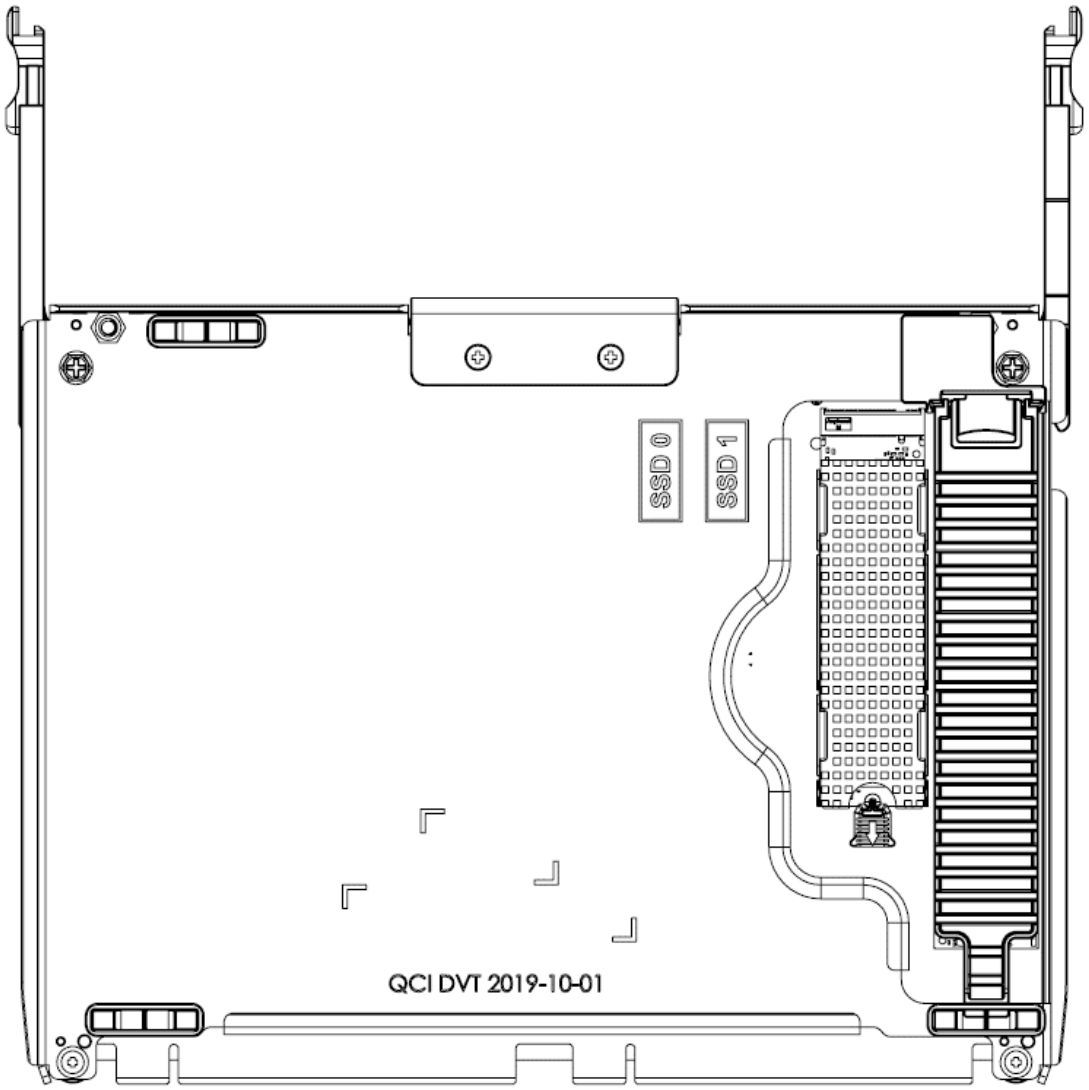

Figure 8-5: Back view of North Dome 1S Server Assembly

9 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 ROTOR or one FAN in each thermal zone failed

9.1 Data Center Environmental Conditions

This section outlines Facebook data center operational conditions.

9.1.1 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.

9.1.2 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.

9.1.3 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 0.005 inches H_2O with a single fan (or rotor) failure.

9.1.4 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.

9.2 Server Operational Conditions

9.2.1 Inlet Temperature

The inlet air temperature will vary. The cooling system should cover inlet temperatures 20°C, 25°C, 30°C, and 35°C. Cooling above 30°C is beyond the Facebook operational condition, but used during validation to demonstrate the thermal reliability and design margin. Any degraded performance is not allowed over the validation range 0°C-35°C.

9.2.2 Pressurization

Except for the condition when one rotor in a server fan fails, the thermal solution should not consider extra airflow from data center cooling fans. If and only if one rotor in a server fan fails, the negative or positive DC pressurization can be considered in the thermal solution in the hot aisle or the cold aisle, respectively.

9.2.3 Fan Redundancy

System should support N+1 fan redundancy. In this design, single roter fan has been used, therefore, under single fan failure, system performance should not be impacted within normal operating temperature range.

9.2.4 Delta T and System Volumetric Flow

The Delta T is the air temperature difference across the system, or the temperature difference between the outlet air temperature and the inlet air temperature. The desired Delta T is greater than 13.9°C (24.5°F).

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 desired airflow is 0.13CFM/W in the system level at sea level up to 30C to meet the Facebook data center operation requirement.

9.2.5 Thermal Margin

The thermal margin is the difference between the maximum theoretical safe temperature and the actual temperature. Unless specified, the system should operate at an inlet temperature of 35°C (95°F) outside of the system with a minimum 4% thermal margin for inlet temperature above 30C or 7% thermal margin for inlet temperatures up to 30°C.

9.2.6 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.

9.2.7 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.

9.3 CPU cooling solution

9.3.1 Orientation

The CPU cooling solution must be designed such that performance is not impacted by server card orientation. All vapor chamber, or heat pipe components must operate in any orientation.

9.3.2 Solution Footprint

The CPU cooling solution must fit within the available space on the server card. The detailed CPU keep out zones can be referred to the mechanical drawing and spec. 9-1The total height of the solution and CPU stack up must be less than 45mm. The CPU cooling solution must not interfere with any tall electrical components on the server card.

9.3.3 9-1Passive Cooling Solution

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

9.3.4 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 must be provided to meet this MTBF.

9.4 M.2 Cooling Solution

Each card contains one boot drive M.2 on the bottom side of the card.

9.4.1 Bottom Side M.2 Cooling Requirements

The cooling solution for M.2 cooling must be thermally optimized design at the lowest cost. Passive cooling is required with a gap pad and heat sink solution being desired. The solution must maintain all M.2 component temperatures within their operational limits during all stress conditions. 5.8mm M.2 connectors must be used to allow spacing for gap pads between M.2 and server card. The total solution stack-up must be 12mm or shorter.

9.4.2 Gap Pad Requirements

The gap pads must be soft and elastic as the application will be between two PCBs. The gap pads must not leave any residue when replaced. The Gap pads must have a thermal conductivity of 2 W/m-k or higher. Gap pad thickness is determined by compression vs pressure that gives the lowest loading force while supporting tolerance of components sandwiching the gap pad. Gap pad AVL provided as needed.

9.4.3 Bottom Side boot M.2 Requirements

Bottom side boot M.2 must maintain operation conditions for all components under all stress conditions. The bottom side boot M.2 is downstream from the high-speed M.2s in standard orientation and may experience higher temperatures due to pre-heating from upstream M.2s. An integrated heat sink with gap pad solution may be required in order to meet bottom side boot M.2 cooling requirements.

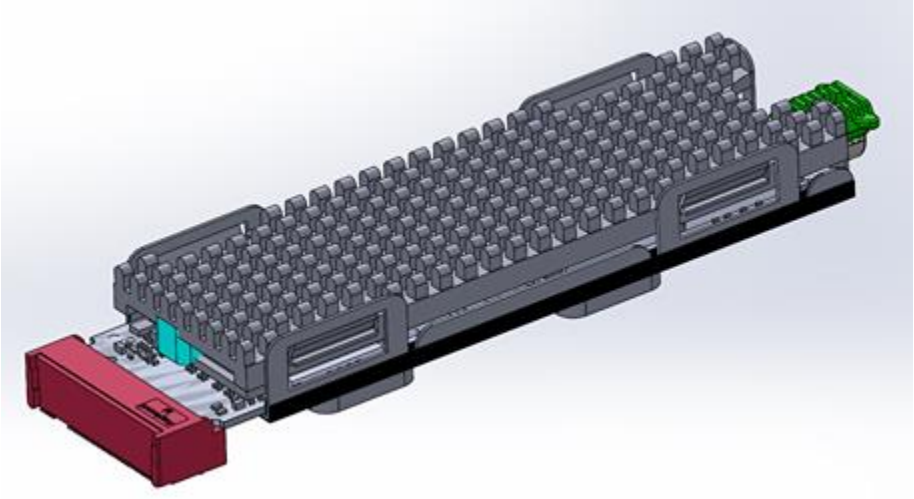

Figure 9-2: Boot drive M.2 with integrated heat sink

9.5 Temperature and Power Sensors

Each card must provide following sensors:

- Temperature sensors for SOC, DIMM, voltage regulators and other critical chips
- Power sensors for the SOC and the whole card
- Voltage sensors for all voltage rails
- 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 of sensor accuracy is +/-1°C typical.

9-39-3

9.6 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.

10 Electrical

10.1 Design Guidelines

Refer to AMD Dev Hub Rome & Milan Motherboard Design Guide (MDG) for design guidelines.

10.2 Electrical Connections of North Dome 1S Server Card

The North Dome 1S server uses both primary and extension edge connectors as defined below.

10.2.1 Primary X16 Edge Connector A

High level pins functions:

- Card Type detection
- Power enable
- UART
- Ejector Latch Detection
- USB2.0
- I2C
- PCIe resets
- PCIe reference clocks

- 4+8 PCIe Gen3 lanes

Note: 7th PCIe Gen3 x4 link, does not have its own PCIe reference clock and PCIe reset signal.

Table 10-1: North Dome 1S Server Primary X16 Edge Connector A Pin-Out

Default Pin-Out			
Pin Name	**B Side**	**A Side**	**Pin Name**
P12V	1	1	PRSNT_A#
P12V	2	2	P12V
P12V	3	3	P12V
GND	4	4	GND
I2C_SCL	5	5	CARD_TYPE
I2C_DATA	6	6	DEV_PWR_EN
GND	7	7	COM_TX
PWR_BTN#	8	8	COM_RX
USB_P	9	9	EJCT_LATCH_DET_N
USB_N	10	10	RESET_BMC
SYS_RESET#	11	11	PCIE_0_RESET#
I2C_ALERT#	12	12	GND
GND	13	13	PCIE_0_REFCLK_P
GND	14	14	PCIE_0_REFCLK_N
PCIE_X4A_TX0_P	15	15	GND
PCIE_X4A_TX0_N	16	16	GND
GND	17	17	PCIE_X4A_RX0_P
GND	18	18	PCIE_X4A_RX0_N
PCIE_X4A_TX1_P	19	19	GND
PCIE_X4A_TX1_N	20	20	GND
GND	21	21	PCIE_X4A_RX1_P
GND	22	22	PCIE_X4A_RX1_N
PCIE_X4A_TX2_P	23	23	GND
PCIE_X4A_TX2_N	24	24	GND
GND	25	25	PCIE_X4A_RX2_P
GND	26	26	PCIE_X4A_RX2_N

PCIE_X4A_TX3_P	27	27	GND
PCIE_X4A_TX3_N	28	28	GND
GND	29	29	PCIE_X4A_RX3_P
GND	30	30	PCIE_X4A_RX3_N
PCIE_X1_TX_P	31	31	GND
PCIE_X1_TX_N	32	32	GND
GND	33	33	PCIE_X1_RX_P
GND	34	34	PCIE_X1_RX_N
PCIE_1_REFCLK_P	35	35	GND
PCIE_1_REFCLK_N	36	36	GND
GND	37	37	PCIE_2_REFCLK_P
GND	38	38	PCIE_2_REFCLK_N
PCIE_1_RESET#	39	39	GND
PCIE_2_RESET#	40	40	GND
GND	41	41	FAST_THROTTLE_N
GND	42	42	NC
NC	43	43	GND
NC	44	44	GND
GND	45	45	NC
GND	46	46	NC
NC	47	47	GND
NC	48	48	GND
GND	49	49	PCIE_X8A_RX0_P
GND	50	50	PCIE_X8A_RX0_N
PCIE_X8A_TX0_P	51	51	GND
PCIE_X8A_TX0_N	52	52	GND
GND	53	53	PCIE_X8A_RX1_P
GND	54	54	PCIE_X8A_RX1_N
PCIE_X8A_TX1_P	55	55	GND
PCIE_X8A_TX1_N	56	56	GND
GND	57	57	PCIE_X8A_RX2_P
GND	58	58	PCIE_X8A_RX2_N
PCIE_X8A_TX2_P	59	59	GND

PCIE_X8A_TX2_N	60	60	GND
GND	61	61	PCIE_X8A_RX3_P
GND	62	62	PCIE_X8A_RX3_N
PCIE_X8A_TX3_P	63	63	GND
PCIE_X8A_TX3_N	64	64	GND
GND	65	65	PCIE_X8B_RX0_P
GND	66	66	PCIE_X8B_RX0_N
PCIE_X8B_TX0_P	67	67	GND
PCIE_X8B_TX0_N	68	68	GND
GND	69	69	PCIE_X8B_RX1_P
GND	70	70	PCIE_X8B_RX1_N
PCIE_X8B_TX1_P	71	71	GND
PCIE_X8B_TX1_N	72	72	GND
GND	73	73	PCIE_X8B_RX2_P
GND	74	74	PCIE_X8B_RX2_N
PCIE_X8B_TX2_P	75	75	GND
PCIE_X8B_TX2_N	76	76	GND
GND	77	77	PCIE_X8B_RX3_P
GND	78	78	PCIE_X8B_RX3_N
PCIE_X8B_TX3_P	79	79	GND
PCIE_X8B_TX3_N	80	80	GND
GND	81	81	P12V
GND	82	82	P12V

10.2.2 Secondary X16 Edge Connector B

The North Dome 1S server also implements an extension x16 edge connector to bring out additional x16 PCIe lanes (pin assignments shown in Table 10-2: North Dome 1S Server Extension X16 Edge Connector B Pin-Out). This extension x16 Edge connector is referred to as Connector B.

Table 10-2: North Dome 1S Server Extension X16 Edge Connector B Pin-Out

Default Pin-Out			
Pin Name	**B Side**	**A Side**	**Pin Name**
P12V	1	1	PRSNT_B#

P12V	2	2	P12V
P12V	3	3	P12V
GND	4	4	GND
NC	5	5	NC
NC	6	6	NC
NC	7	7	NC
NC	8	8	GND
NC	9	9	PCIE_4_REFCLK_P
GND	10	10	PCIE_4_REFCLK_N
PCIE_3_RESET#	11	11	GND
PCIE_4_RESET#	12	12	GND
PCIE_5_RESET#	13	13	PCIE_5_REFCLK_P
GND	14	14	PCIE_5_REFCLK_N
PCIE_X16A_TX0_P	15	15	GND
PCIE_X16A_TX0_N	16	16	GND
GND	17	17	PCIE_X16A_RX0_P
GND	18	18	PCIE_X16A_RX0_N
PCIE_X16A_TX1_P	19	19	GND
PCIE_X16A_TX1_N	20	20	GND
GND	21	21	PCIE_X16A_RX1_P
GND	22	22	PCIE_X16A_RX1_N
PCIE_X16A_TX2_P	23	23	GND
PCIE_X16A_TX2_N	24	24	GND
GND	25	25	PCIE_X16A_RX2_P
GND	26	26	PCIE_X16A_RX2_N
PCIE_X16A_TX3_P	27	27	GND
PCIE_X16A_TX3_N	28	28	GND
GND	29	29	PCIE_X16A_RX3_P
GND	30	30	PCIE_X16A_RX3_N
PCIE_X16B_REFCLK_P	31	31	GND
PCIE_X16B_REFCLK_N	32	32	GND

GND	33	33	PCIE_X16B_RX0_P
GND	34	34	PCIE_X16B_RX0_N
PCIE_X16B_TX0_P	35	35	GND
PCIE_X16B_TX0_N	36	36	GND
GND	37	37	PCIE_X16B_RX1_P
GND	38	38	PCIE_X16B_RX1_N
PCIE_X16B_TX1_P	39	39	GND
PCIE_X16B_TX1_N	40	40	GND
GND	41	41	PCIE_X16B_RX2_P
GND	42	42	PCIE_X16B_RX2_N
PCIE_X16B_TX2_P	43	43	GND
PCIE_X16B_TX2_N	44	44	GND
GND	45	45	PCIE_X16B_RX3_P
GND	46	46	PCIE_X16B_RX3_N
PCIE_X16B_TX3_P	47	47	GND
PCIE_X16B_TX3_N	48	48	GND
GND	49	49	PCIE_X16C_RX0_P
GND	50	50	PCIE_X16C_RX0_N
PCIE_X16C_TX0_P	51	51	GND
PCIE_X16C_TX0_N	52	52	GND
GND	53	53	PCIE_X16C_RX1_P
GND	54	54	PCIE_X16C_RX1_N
PCIE_X16C_TX1_P	55	55	GND
PCIE_X16C_TX1_N	56	56	GND
GND	57	57	PCIE_X16C_RX2_P
GND	58	58	PCIE_X16C_RX2_N
PCIE_X16C_TX2_P	59	59	GND
PCIE_X16C_TX2_N	60	60	GND
GND	61	61	PCIE_X16C_RX3_P
GND	62	62	PCIE_X16C_RX3_N
PCIE_X16C_TX3_P	63	63	GND
PCIE_X16C_TX3_N	64	64	GND
GND	65	65	PCIE_X16D_RX0_P

GND	66	66	PCIE_X16D_RX0_N
PCIE_X16D_TX0_P	67	67	GND
PCIE_X16D_TX0_N	68	68	GND
GND	69	69	PCIE_X16D_RX1_P
GND	70	70	PCIE_X16D_RX1_N
PCIE_X16D_TX1_P	71	71	GND
PCIE_X16D_TX1_N	72	72	GND
GND	73	73	PCIE_X16D_RX2_P
GND	74	74	PCIE_X16D_RX2_N
PCIE_X16D_TX2_P	75	75	GND
PCIE_X16D_TX2_N	76	76	GND
GND	77	77	PCIE_X16D_RX3_P
GND	78	78	PCIE_X16D_RX3_N
PCIE_X16D_TX3_P	79	79	GND
PCIE_X16D_TX3_N	80	80	GND
GND	81	81	P12V
POWER_FAIL_N	82	82	P12V

10.3 Pin Definition

Table 10-3: Detailed Pin Definitions provides a detailed explanation of the pins. The direction of the signals is always defined from the perspective of the North Dome 1S Server module.

Table 10-3: Detailed Pin Definitions

Pin	Direction	Required/ Configurable	Pin Definition
P12V	Input	Required	12VAUX power from platform
I2C_SCL	Input/Output	Required	I^2C clock signal. I^2C is the primary sideband interface for server management functionality. 3.3VAUX signal. Pull-up is provided on the platform.
I2C_SDA	Input/Output	Required	I^2C data signal. I^2C is the primary sideband interface for server management functionality. 3.3VAUX signal. Pull-up is provided on the platform.

I2C_ALERT#	Output	Required	I^2C alert signal. Alerts the BMC that an event has occurred that needs to be processed. 3.3VAUX signal. Pull-up is provided on the platform.
PWR_BTN#	Input	Required	Power on signal. When driven low, it triggers the server to begin its power-on sequence transition from ACPI S5 state towards S0 state. 3.3VAUX signal. Pull-up is provided on the platform. If PWR_BTN# is held low for shorter than 4 seconds, then this indicates a soft (graceful) power off. Otherwise, a hard shutdown is initiated.
SYS_RESET#	Input	Required	System reset signal. When driven low, it triggers the server to start warm reboot process. 3.3VAUX signal. Pull-up is provided on the platform.
PRSNT_A#	Output	Required	Present signal. This is pulled low on the card to indicate that a card is installed. 3.3VAUX signal. Pull-up is provided on the platform.
PRSNT_B#	Output	Required	Extension edge connector Present signal. This is pulled low on the card to indicate that a card is installed. 3.3VAUX signal. Pull-up is provided on the platform.
COM_TX	Output	Required	Serial transmit signal. Data is sent from the server card to the BMC. 3.3VAUX signal.
COM_RX	Input	Required	Serial receive signal. Data is sent from the BMC to server card. 3.3VAUX signal.
CARD_TYPE (GPIO0)	Output	Required	CARD_TYPE is an output signal to inform platform that if this card is a server or a device. For a server, this pin should be tied to GND through a 10K resistor on the server card.
DEV_PWR_EN (GPIO1)	Output	Required	DEV_PWR_EN is an output signal from server card to enable active/main/S0 power on device card. It is useful when 1S server is going through power cycling or AC cycling to align server and devices power domain. Active high, 3.3VAUX signal.
EJCT_LATCH_DET_N (GPIO2)	Output	Required	EJCT_LATCH_DET_N is an output signal to indicate the server card is fully seated with ejector latch closed

			and ready for power on. Platform designer can use this signal to control the 12V power to server card and avoid surprise 1S server removal from a hot slot. Active low, 3.3VAUX signal, pull-up should be provided on the platform.
RESET_BMC (GPIO3)	Output	Required	RESET_BMC is an output signal to reset BMC on the platform. Active high, 3.3VAUX signal, pull-down on the platform.
PCIE_[0:5]_RESET#	Output	Required	PCIe reset signal. If a PCIe bus is connected, this signal provides the reset signal indicating the card VRs and clocks are stable when driven high to 3.3V.
PCIE_[0:5]_ REFCLK_P/N	Output	Required	PCIe reference clock. This signal may or may not be connected on the platform.
PCIE_X4A_TX[0:3]_P/N	Output	Required	PCIe x4 bus transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X4A_RX[0:3]_P/N	Input	Required	PCIe x4 bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X8A_TX[0:3]_P/N	Output	Required	PCIe x8 1st nibble bus transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X8A_RX[0:3]_P/N	Input	Required	PCIe x8 1st nibble bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X8B_TX[0:3]_P/N	Output	Required	PCIe x8 2nd nibble bus transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X8B_RX[0:3]_P/N	Input	Required	PCIe x8 2nd nibble bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X16A_TX[0:3]_P/N	Output	Required	PCIe x16 1st nibble bus transmit signals. Data is sent from the server card to the platform. These signals

			may or may not be connected on the platform.
PCIE_X16A_RX[0:3]_P/N	Input	Required	PCIe x16 1st nibble receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X16B_TX[0:3]_P/N	Output	Required	PCIe x16 2nd nibble transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X16B_RX[0:3]_P/N	Input	Required	PCIe x16 2nd nibble bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X16C_TX[0:3]_P/N	Output	Required	PCIe x16 3rd nibble bus transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X16C_RX[0:3]_P/N	Input	Required	PCIe x16 3rd nibble bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X16D_TX[0:3]_P/N	Output	Required	PCIe x16 4th nibble bus transmit signals. Data is sent from the server card to the platform. These signals may or may not be connected on the platform.
PCIE_X16D_RX[0:3]_P/N	Input	Required	PCIe x16 4th nibble bus receive signals. Data is sent from the platform to the server card. These signals may or may not be connected on the platform.
PCIE_X1_TX_P/N	Output	Required	PCIe x1 transmit signals. Data is sent from the 1S Server module to the platform. These signals may or may not be connected on the platform.
PCIE_X1_RX_P/N	Input	Required	PCIe x1 receive signals. Data is sent from the platform to the 1S Server module. These signals may or may not be connected on the platform.
USB_P/N	Input/Output	Required	USB 2.0 differential pair.
FAST_THROTTLE_N	Input	Required	Active low open drain signal with pull-up on 1S server. Platform generates this signal and uses it as a

			big hammer to throttle 1S server down to lowest possible power state as fast as possible.
POWER_FAIL_N	Input	Required	Active low open drain signal with pull-up on 1S server. When this signal is asserted by platform, it informs 1S server that base system is going to cut 12V power to 1S server in certain amount of time, which is pre-defined by base system. It is possible for 1S server to perform graceful shutdown based on this signal.

10.4 Ethernet

The North Dome 1S server doesn't support Ethernet. A separate network interface card (NIC) recommended for network access.

10.5 NIC Sideband

As North Dome 1S server card doesn't support SoC integrated ethernet, there is no requirement for NIC sideband.

10.6 USB

The USB connection is one USB 2.0 port.

10.7 Serial port

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

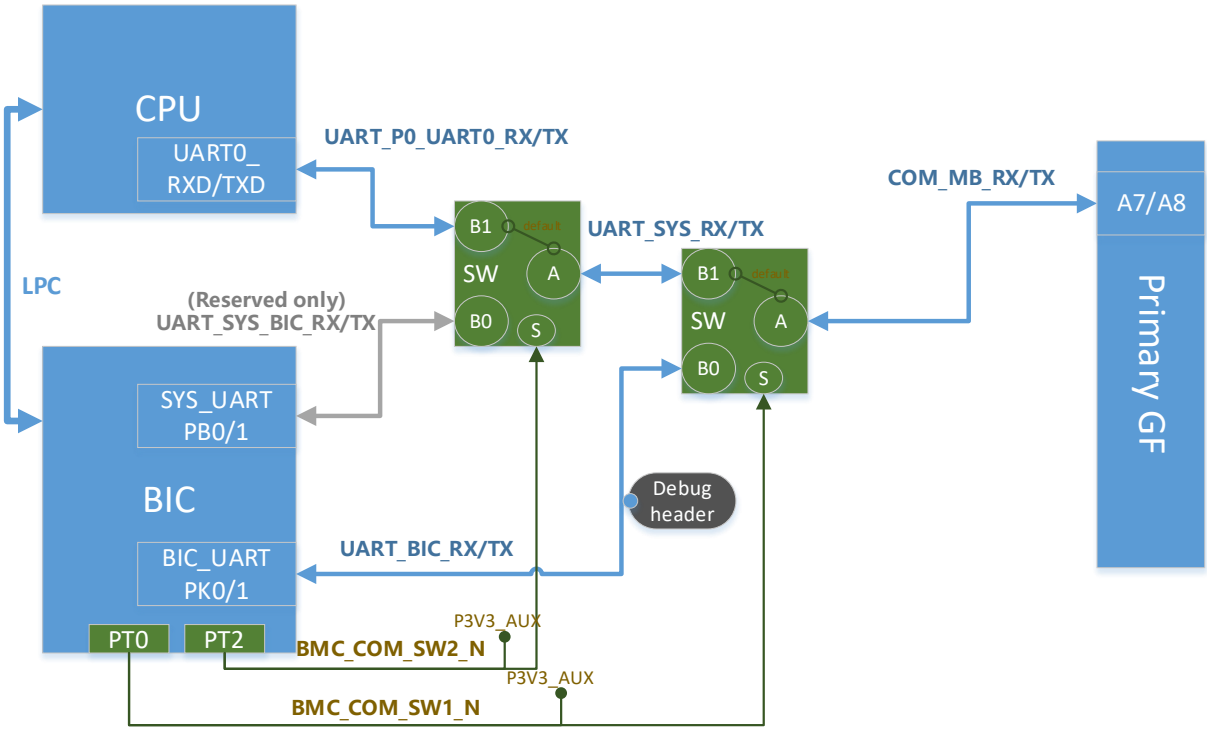

Figure 10-1: North Dome UART topology

10.8 I²C

A single I²C connection is used to connect the BMC on the platform to the Bridge IC on the 1S server as the primary server management interface. It shall support 1MHz or higher. The I²C alert signal is required and is used as an interrupt for both the Bridge IC and the BMC.

Both the BMC and the Bridge IC are I²C master devices on the bus and they communicate with each other via the Intelligent Platform Management Bus (IPMB) protocol. To achieve maximum bandwidth and avoid conflicts, no other devices should use this bus except for the BMC and the Bridge IC.

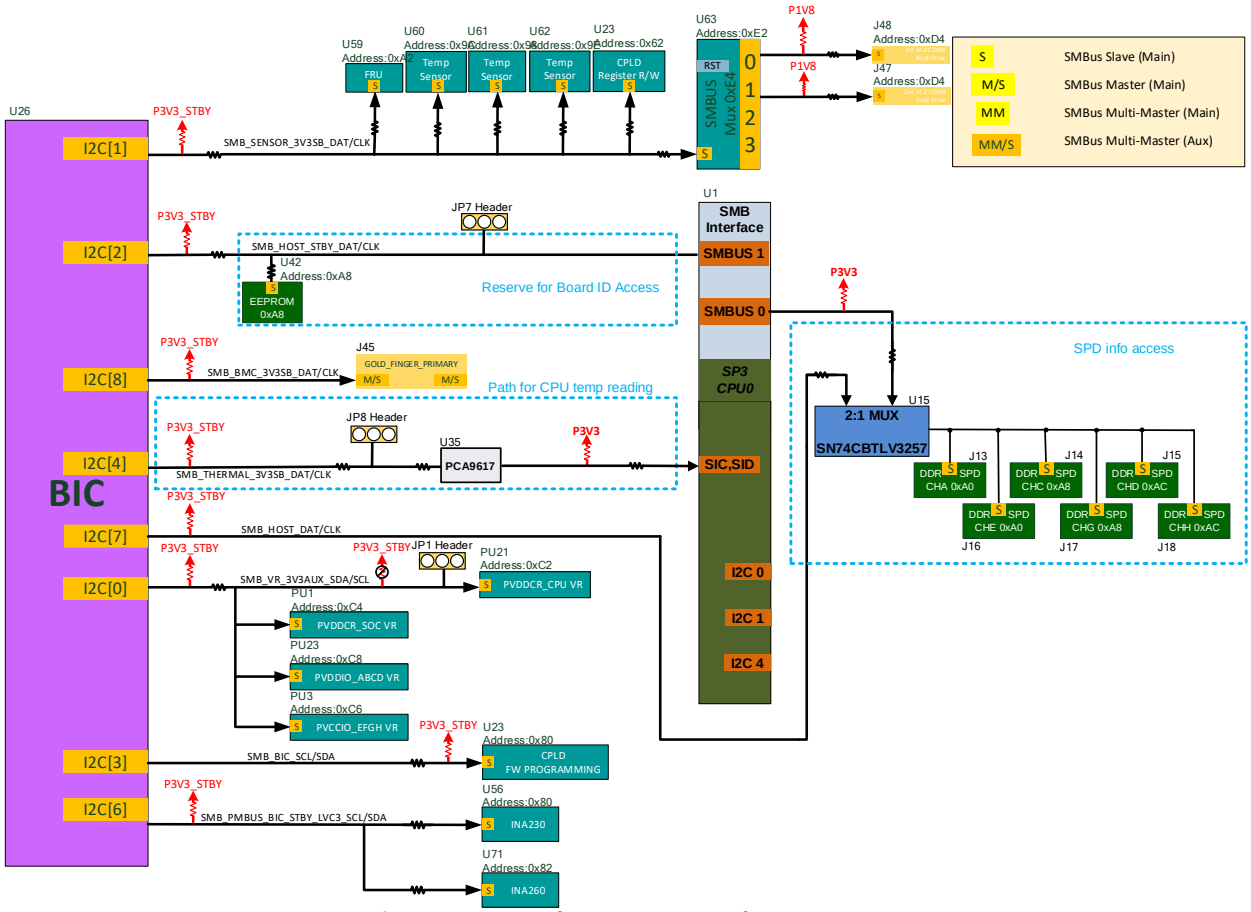

Figure 10-2: North Dome I2C topology

10.9 SPI

North Dome is using SPI bus for CPU to read/write BIOS firmware from SPI flash. On top of that, the SPI flash can also be read/write thru Bridge-IC from out-of-band access.

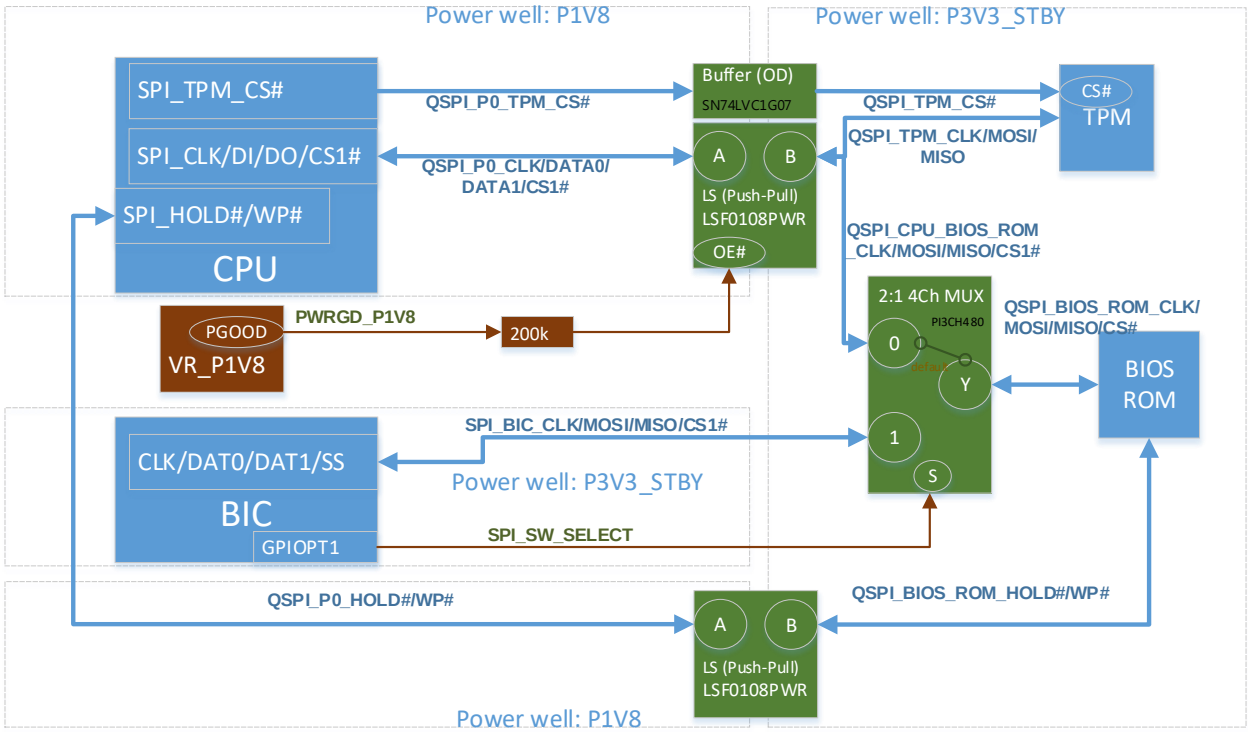

Figure 10-3: North Dome SPI topology

10.10 JTAG

HDT is the primary JTAG port to North Dome server board to communicate with CPU. The BIC is also on the bus and can take over the bus to become the master by flipping a physical switch

A second JTAG connection is available between BIC and CPLD.

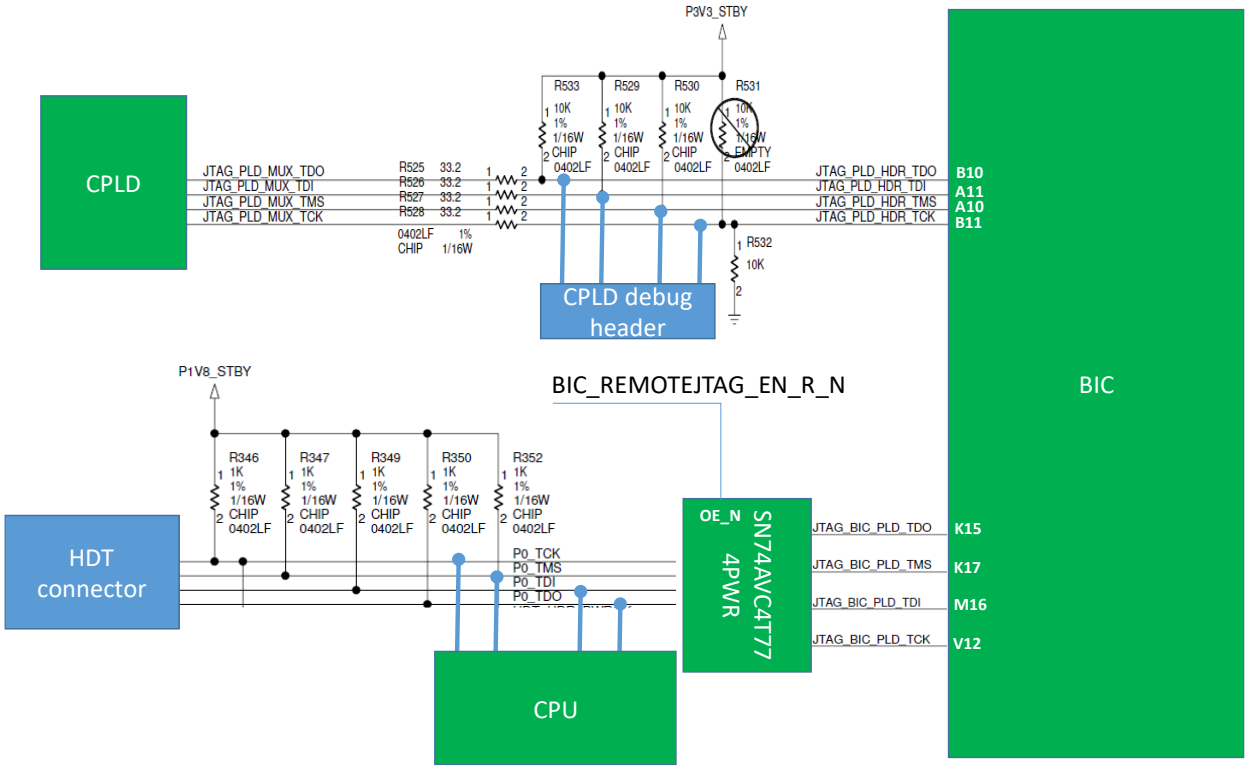

Figure 10-4: North Dome JTAG topology

10.11 Slot ID and GPIO

Due to the nature of the 1S server architecture, a BMC is always preferred on the platform side to work with the 1S server. When a BMC is present in the system, the Bridge IC shall always request its slot ID from the BMC but not probe the slot ID by itself.

There are four slot ID bits defined in 1S server spec. These pins can be used as GPIO pins if they are not used as slot IDs.

North Dome and Twin Lakes 1S server has the same definition 4 GPIO pins with specific functions. Please refer to pin definition in Table 10-3: Detailed Pin Definitions for details.

11 Power

11.1 Input Voltage

Power for the card is provided via seven 12V pins on the primary connector and seven more 12V pins on the extension connector. Each pin supports a maximum 1.1A of current.

The nominal 12V input voltage is defined as 12.5V, +/-7%.

11.2 North Dome 1S server power capacity

The North Dome 1S server's maximum power is up to 180W DC with full configuration. It is critical to develop a sophisticated thermal solution for the platform to keep the North Dome 1S server operating at a safe condition when it runs with a maximum power load.

The designer also should pay attention to SOC peak power requirement to ensure when North Dome being delivered together with platform it operates coherently.

11.3 Power sequence and standby power

Because there is only one 12V power input to the 1S server, there is not a power sequence requirement to power on the North Dome 1S server card from the platform perspective. However, a standby 3.3V_AUX power rail on the card is required to power the Bridge IC at all power states. The SoC also requires some stand-by power rails. A power CPLD that controls the power sequencing of the entire server is also powered by the 3.3V_AUX standby power rail.

It is the designer's responsibility to provide proper standby power rails from the 12V_AUX input with possible specific power sequencing. Care must be taken to avoid any leakage path among the power domains including 1S server stand-by, 1S server core, and baseboard platform power.

11.4 Capacitive Load

The capacitance on the input 12V rail of the North Dome 1S server shall be lower than 3200uF to be compatible with Yosemite V2 platform.

11.5 VR Efficiency

All Voltage Regulators (VRs) providing over 10W on the card are at least 90% efficient when loaded between 30% and 90% of the full load.

11.6 Power Reading and Capping

The North Dome 1S server implements sophisticated power management features.

The North Dome 1S server shall have the power monitoring capability to read power consumption reliably and accurately and can report a one-second average power reading with 1% accuracy. As shown in Figure 6-1: North Dome 1S Server Block Diagram, a power sensor with

I²C interface is used at the 12.5V power input to the card and is readable by the Bridge IC. Both the BMC and SoC can obtain the whole-card power consumption information from the Bridge IC.

The North Dome 1S server shall throttle itself down to lowest possible power state within defined response time when the platform asserts the FAST_THROTTLE_N signal or it receives request from BMC, or an overpower event reported by on-card power monitor. The response time should be tunable by management firmware.

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 platform can 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 a 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.

12 North Dome 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 I²C bus as the management interface. This section identifies the required information that must be accessible from the BMC.

12.1 Bridge IC

The North Dome 1S server uses a Bridge IC is defined as the bridging device between the CPU and the BMC. The Bridge IC is on stand-by power so that it can be accessed by the BMC even when the CPU is powered down.

On the platform side, the BMC and Bridge IC communicate with each other with IPMI messages over the I²C bus. To enable prompt communication, this I²C bus shall be a point-to-point link without any other devices on the same bus. It shall run in high-speed mode with a minimum speed of 400KHz. When possible, a 1MHz or better speed is strongly recommended.

On the 1S server side, the Bridge IC has FRU EEPROM and thermal sensors on a local I²C 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 8.5. 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 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 North Dome 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. This microcontroller must have two Serial Peripheral Interfaces (SPI). The first SPI bus is used as its own boot ROM if it does not have an integrated boot ROM. The second SPI can be used to reprogram the SoC's boot ROM when it is corrupted with a multiplexer and BMC's support. The Bridge IC on the North Dome 1S server is a Texas Instrument's Tiva microcontroller.

12.2 I^2C addressing

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

12.3 Message Transfer

As the bridge between the server CPU 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 CPU's Keyboard Controller Style (KCS) request to the BMC and then send the received response back to the SoC.

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

12.4 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.

12.5 POST Code Access

During the power-on stage, the CPU 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.

12.6 IPMB Interface

The Bridge IC provides an IPMB interface for the BMC to access various IPMI resources on the North Dome 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.

12.7 Firmware Update

The Bridge IC can update firmware of the programmable devices on the North Dome 1S Server, such as BIOS, 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 North Dome 1S Server, as well as detect and retransmit corrupted firmware image packets during transit from the BMC to the Bridge IC.

12.8 Network status LED Control

Any network status LEDs will be contained on the NIC. North Dome will not implement network status LEDs.

12.9 GPIO Register

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 four bytes to represent 32 signals that indicate various conditions as shown in Table 11-1: Bridge IC GPIO table.

Table 11-1: Bridge IC GPIO table

No.	GPIO offset	GPIO Pin Function	Comments
1	Byte 1 -bit [0]	Power Good – CPU Core	Indicates that the CPU's core power input is good
2	Byte 1 - bit [1]	Power Good – CPLD core	Indicates that the CPLD's core power input is good
3	Byte 1 - bit [2]	North DDR Channel Voltage regulator hot	Indicates that the DDR Channel C/D Voltage Regulator is hot
4	Byte 1 - bit [3]	South DDR Channel Voltage regulator hot	Indicates that the DDR Channel G/H Voltage Regulator is hot
5	Byte 1 - bit [4]	Inlet and Outlet Temperature Alert	Indicates an Inlet or Outlet over temperature
6	Byte 1 - bit [5]	CPU Throttle	System firmware request CPU throttle
7	Byte 1 - bit [6]	CPU Voltage regulator hot	Indicates that the CPU Voltage Regulator is hot
8	Byte 1 - bit [7]	CPU Voltage regulator hot	Indicates that the CPU Voltage Regulator is hot
9	Byte 2 - bit [0]	CPU thermal trip	Indicates that CPU experienced over temperature event and shut-down
10	Byte 2 - bit [1]	BIC ready	Indicated BIC is ready
11	Byte 2 - bit [2]	CPU Alert	Indicate that CPU has some events
12	Byte 2 - bit [3]	North DDR Channel FIVR Fault	Indicates a DDR Channel C/D Voltage Regulator Error condition

13	Byte 2 - bit [4]	CPU FIVR fault	Indicates a CPU internal Voltage Regulator Error condition
14	Byte 2 - bit [5]	CPU FIVR fault	Indicates a CPU Voltage Regulator Error condition
15	Byte 2 - bit [6]	South DDR Channel FIVR Fault	Indicates a DDR Channel G/H Voltage Regulator Error condition
16	Byte 2 - bit [7]	SMI Active	Indicates system has receive an SMI
17	Byte 3 - bit [0]	Fast Throttle	Indicates system is currently experiencing a fast throttle event
18	Byte 3 - bit [1]	System management interrupt	System management interrupt
19	Byte 3 - bit [2]	Platform Reset	Platform and PCIe Reset
20	Byte 3 - bit [3]	Front Panel Reset Input	Initiate platform reset (front panel reset button being pressed)
21	Byte 3 - bit [4]	Front Panel Reset Output	Host Reset Output from Bridge-IC
22	Byte 3 - bit [5]	Bios Power-on Self Test (POST) complete	Bios Power-on Self Test (POST) complete
23	Byte 3 - bit [6]	Sleep S3 state	When low, indicates CPU has entered S3 state or lower
24	Byte 3 - bit [7]	Power Good CPLD	Indicates the CPLD has completed power sequencing
25	Byte 4 - bit [0]	Boot SPI selection	Select boot SPI0 or SPI1 as boot SPI
26	Byte 4 - bit [1]	Ejector latch detection	Identify ejector fully closed
27	Byte 4 - bit [2]	BMC Reset	Reset BMC from host
28	Byte 4 - bit [3]	Host / Bridge-IC UART select	Select Host UART or Bridge-IC Debug UART (2)
29	Byte 4 - bit [4]	BMC ready	Indicated BMC is ready
30	Byte 4 - bit [5]	Host / Bridge-IC UART select	Select Host UART or Bridge-IC Debug UART (1)
31	Byte 4 - bit [6]	I2C MUX reset	Reset I2C MUX
32	Byte 4 - bit [7]	At-scale-debug (ASD) PREQ	CPU probe mode request
33	Byte 5 - bit [0]	At-scale-debug (ASD) JTAG TRST	JTAG Reset
34	Byte 5 - bit [1]	System Throttle	Indicates system is currently throttling
35	Byte 5 - bit [2]	Input Power Alert	Indicates an input power voltage/current fault
36	Byte 5 - bit [3]	M.2 Power Alert	Indicates voltage current fault on 3.3V for M.2
37	Byte 5 - bit [4]	Host Power Button Out	Power button indicates from BMC
38	Byte 5 - bit [5]	HDT Present	Indicates HDT debug connector is present in system
39	Byte 5 - bit [6]	North DDR Channel event	Indicates a DDR Channel C/D Event
40	Byte 5 - bit [7]	South DDR Channel event	Indicates a DDR Channel G/H Event

41	Byte 6 - bit [0]	FP Power Button	FP Power Button Input
42	Byte 6 - bit [1]	Reset RTC	Clear the RTC
43	Byte 6 - bit [2]	BIOS Recovery Request	Indicates BIOS should enter recovery mode
44	Byte 6 - bit [3]	CPU All Power Good	Indicates CPU has all good power rails and has come out of reset
45	Byte 6 - bit [4]	Power LED	LED output to indicate Power
46	Byte 6 - bit [5]	Fault LED	LED output to indicate fault on server
47	Byte 6 - bit [6]	DIMM MUX Select	Used to select DIMM sensor to access via mux
48	Byte 6 - bit [7]	Caterr Error	Indicates Catostrophic Error Occurred

12.10 IPMI commands

The Bridge IC must support the IPMI commands shown in Table 11-2: Bridge IC supported IPMI command.

Table 11-2: Bridge IC supported IPMI command

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
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 Bridge IC configuration	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
Request 4-Byte Post Buffer Data	OEM (0x38)	2Ah
Read APML Data	OEM (0x38)	2Ch
Write APML Data	OEM (0x38)	2Dh
Request APML Mailbox Data	OEM (0x38)	2Eh
Read APML Mailbox Data	OEM (0x38)	2Fh
Rest BIC	OEM (0x38)	32h
Set System GUID	OEM (0x38)	EFh

Table 11-3: Bridge IC supported OEM specific IPMI command provides details of the IPMI Original Equipment Manufacturer (OEM) commands that are defined in Table 11-2: Bridge IC supported IPMI command.

Table 11-3: Bridge IC supported OEM specific IPMI command

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: 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: 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: 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: 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 Manageability Engine and get the response from 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 5:10 – 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 4:9 – GPIO enable mask, refer to GPIO mapping table 0b: Disable 1b: Enable Byte 10:15 – 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 Buffer 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.
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 –target 01h: CPLD 02h: Bridge IC 03h: Bridge IC bootloader 04h: PVDDCR_CPU VR 05h: PVDDCR_SOC VR 06h: PVDDIO_ABCD VR 07h: PVDDIO_EFGH 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. VR version (Hexadecimal) – VR CRC values, 4 bytes length. Ex: 0x7B3CEC91 Bridge IC bootloader version (Decimal) – 2 bytes length, version 1.08 Return data will be 0x01 0x08 Byte X+1 – VR Update Available Number	
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
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	
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.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
12h	Request POST Buffer 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.
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.
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	

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
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	This command is used to read BIOS image then response to BMC
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	This command is used to get BIOS flash size then response to BMC

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
25h	Clear CMOS	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	
2Ah	Request 4- Byte POST Buffer Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Page (Start From 01h) 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	Same as 'Request Post Buffer Data' but in 4- Byte
2Bh	Write BIOS image	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 –target 00h: BIOS Byte 5:8 – Offset Byte 9 – Data length (up to 0xF0) Data 10:X 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 write BIOS image then response to BMC.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
2Ch	Read APML Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Request Type 0x00 : RMI 0x01 : TSI 0x02: MCA 0x03: CPUID RMI, TSI: Byte 5 – Register Address MCA: Byte 5 – Thread Address Byte 6:9 – Register Address (LSB first) CPUID: Byte 5 – Thread Address Byte 6:9 – Register Address (LSB first) Byte 10 – ECX Value RMI EX: 0x15 0xA0 0x00 0x00 0x00 TSI EX: 0x15 0xA0 0x00 0x01 0x01 MCA EX: 0x15 0xA0 0x00 0x02 0x00 0x01 0x20 0x00 0xC0 CPUID EX: 0x15 0xA0 0x00 0x03 0x00 0x0D 0x00 0x00 0x00 0x10 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 RMI TSI: Byte 5 – Data MCA: CPUID: Byte 5 – How many bytes return Byte 6 – Status Code Byte 7:14 – Data	Request data via AMD APML Thread Address: between 0x00 and 0x3F ECX Value: ECX[3:0]_000Xb ECX[3:0] is the initial ECX value for extended CPUID operations. Must be 0h for non-extended operations. X: 0b=Return ebx:eax; 1b=Return edx:ecx.

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
2Dh	Write APML Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Request Type 0x00 : RMI 0x01 : TSI RMI, TSI: Byte 5 – Register Address Byte 6 – 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	Write data via AMD APML
2Eh	Request APML Mailbox Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – APML Mailbox Command Byte 5:8 – APML Mailbox Data (LSB first) EX: 0x15 0xA0 0x00 0x01 0x00 0x00 0x00 0x00 (Read Package Power Consumption) EX: 0x15 0xA0 0x00 0x03 0x00 0x00 0x00 0x00 (Read Package Power Limit) EX: 0x15 0xA0 0x00 0x02 0xF8 0x24 0x01 0x00 (Write Package Power Limit 75000 mWatts) 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 ID (Only Save Last 10 Records)	Request data via AMD APML Mailbox After requesting APML mailbox data, it needs about 20ms to complete the request.
2Fh	Get APML Mailbox Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Request ID (From command 2Eh) 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 If APML Complete: Byte 5 – Response Command Byte 6:9 – Response Data (LSB first)	Get data via AMD APML Mailbox

Net Function = OEM (0x38), LUN = 00			
Code	Command	Request, Response Data	Description
30h	Set Sensor Polling	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – Enable/Disable All Sensor Polling 0x00 : Enable 0x01 : Disable 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	Enable/Disable All Sensor Polling
32h	Reset BIC	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	Reset BIC
3Fh	Request and Get APML Mailbox Data	Request: Byte 1:3 – IANA ID – 00A015h, LS byte first Byte 4 – APML Mailbox Command Byte 5:8 – APML Mailbox Data (LSB first) EX: 0x15 0xA0 0x00 0x01 0x00 0x00 0x00 0x00 (Read Package Power Consumption) EX: 0x15 0xA0 0x00 0x03 0x00 0x00 0x00 0x00 (Read Package Power Limit) EX: 0x15 0xA0 0x00 0x02 0xF8 0x24 0x01 0x00 (Write Package Power Limit 75000 mWatts) 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 If APML Complete: Byte 5 – Response Command Byte 6:9 – Response Data (LSB first)	This command is similar with 0x2E, but it gets the data directly. That means it doesn't need to call 0x2F command.

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)	

12.11 Management interface

The CPU 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 CPU 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.

12.12 Serial Console

The serial console of the North Dome server is used as the BIOS or OS serial console and will also be available as a SOL connection via the 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.

12.13 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.

12.14 Thermal Alerts

The CPU 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 act quickly. The I^2C alert signal must be used. In some cases, an over temperature condition may occur which forces the CPU to power-off immediately. This condition must be logged.

12.15 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)
- CPU 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
- CPU/DIMM Hot
- VR Hot

Event Only Sensors Include:

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

12.16 LEDs

Along the top edge of the card, a blue LED is used to indicate 12V status of the server.

It is possible to include a blinking amber heartbeat LED on the North Dome 1S server to indicate that the Bridge IC is in operating mode for validation phases.

13 System Firmware (Bios)

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

13.1 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

13.2 UEFI

The BIOS shall be a UEFI compatible BIOS.

13.3 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 SoC settings can be tuned to achieve the optimal combination of performance and power consumption.

13.4 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.

13.5 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. Network IPv6
2. Network IPv4
3. SSD flash drive
4. Built-in EFI Shell

This process loops indefinitely and requires no user intervention.

13.6 iSCSI Boot

The BIOS shall be capable of iSCSI network boot.

13.7 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

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 five minutes to complete

13.8 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 ethernet 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

13.9 SMBIOS Event Log

Per the SMBIOS specification Version 2.6, the BIOS implements SMBIOS Type 15 for an event log and the assigned area is large enough to hold more than 500 event records (assuming the maximum event record length is 24 bytes, then the size will be larger than 12KB) and follows the SMBIOS event log organization format for the event log.

A system access interface and application software must be provided to retrieve and clear the event log from the BIOS, including, at minimum, a Linux application for the CentOS operating system and driver as needed. The event log must be retrieved and stored as a readable text file that is easy to handle by a scripting language under Linux. Each event record includes enhanced information identifying the error source device's vendor ID, card slot ID, and device ID.

13.10 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

13.11 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 for mass production. When the threshold is reached, the BIOS logs the event and includes the physical DIMM location.

13.12 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.

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:

- The first hex character indicates which CPU interfaces the DIMM module
- 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.

14 Platform Security

14.1 TPM (Trusted Platform Module)

CPU 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.

15 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

15.1 Vibration and Shock

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

Table 15-1: 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	2G/4G/6G, half sine, 11ms, total 6 shocks, along three axes (+/-)	12G, half sine, 11ms, total 6 shocks, along three axes (+/-)

16 Prescribed Materials

16.1 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

16.2 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.

16.3 Component De-rating

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

17 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

18 Revision History

Author	Description	Revision	Date
Damien Chong Jon Ehlen Yueming Li Ben Wei	▪ Initial draft	0.1	11/16/2018
Damien Chong Jon Ehlen Yueming Li Ben Wei	▪ Mechanical update ▪ Thermal update ▪ Figure & table clean-up	0.2	11/19/2018
Damien Chong	▪ M.2 requirement, electrical requirement, air flow, mechanical diagram update	0.3	1/23/2019
Damien Chong	▪ Minor update on PCB thickness ▪ Update on S&V requirements	0.31	2/21/2019
Damien Chong	▪ Updating with latest changes	0.4	6/7/2019
Todd Westhauser Chenyu Xu	▪ Updating based on DVT Board ▪ Mechanical Diagram update	0.41	9/13/1
Damien Chong	▪ Milan update	0.5	7/24/2020
Damien Chong	▪ OCP spec new template update	0.6	9/8/2021
Damien Chong	▪ Update system management GPIO and IPMI	0.7	9/21/2021

Appendix A - Requirements for IC Approval

List all the requirements in one summary table with links from the sections.

Requirements	Details	Link to which Section in Spec
Contribution License Agreement	OWF CLA OWFa1.0 Final Specification Agreement	License
Tenets	Openness Efficiency Impact	OCP Tenets Compliance
Supplier available within 120 days	Quanta Computer	
Will they apply for OCP product recognition?	Yes	