OpenSequana Blade Interface Specification

Document Properties

Contract Number	101033975
Contractual Deadline	M15
Dissemination Level	Public
Nature	Other
Edited by :	ATOS
Authors	Jamal Nasri (ATOS), François Homps (ATOS)
Reviewers	Sylvie Lesmanne (ATOS)
Date	31/05/25
Keywords	Open Hardware, BullSequana XH300, 100% DLC
Status	Final
Release	1.1

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101033975. The JU receives support from the European Union's Horizon 2020 research and innovation programme and France, Germany, Italy, Greece, United Kingdom, Czech Republic, Croatia.

History of Changes

Release	Date	Author, Organization	Description of Changes
0.1	26/04/2021	P. Bernier-Bruna, ATOS	Template Creation
1.0	03/04/2023	J. NASRI, ATOS	First published version
1.1	31/05/2025	J. NASRI, ATOS F. Homps (ATOS)	Clarifications regarding firmware requirements and use of EUPEX deliverable template.

Table of Contents

List of Figures

List of Tables

1 Executive Summary

This deliverable is related to the Tasks 4.2 and 4.3 of the Work package 4 in EUPEX, which focus on the hardware definition (4.2) and the design (4.3) of nodes based on Rhea1 processors.

In order to reuse European technologies, it has been decided to develop blades compatible with the ATOS OpenSequana program. This program enables any ATOS' partner to access the BullSequana XH3000 interfaces specifications, for him to develop its own compute module (called blade in the BullSequana XH3000 environment). This specification is the entry-point of the OpenSequana program and provides the high-level specification of the blade in order to fit in BullSequanaXH3000 rack. As a design guide, it focuses on the interfaces of the OpenSequana blade. Based on these specifications, the partner can develop its own compute module (called blade in the BullSequana XH3000 environment).

In EUPEX, the Rhea1 node is developed by ATOS, and the Rhea1+accelerator node by one partner. EUPEX is the best opportunity to provide the specification to a first partner and to improve this specification on partner request. Some additional documents providing more detailed information are listed in §11References, but are not part of this deliverable.

The release 1.0 has been delivered on time, but the release 1.1 is an update with some evolution in the Firmware requirements.

This specification document is organized as follows:

- **Chapter 4- BullSequanaXH3000 Overview**
  This chapter introduces the different basic concepts implemented in BullSequanaXH3000 and describes the functional architecture of an OpenSequana module.
- **Chapter 5 - Mechanical interfaces**
  This chapter focuses on the hardware OpenSequana mechanical interfaces.
- **Chapter 6- Thermal interfaces**
  This chapter focuses on the hardware OpenSequana thermal interfaces.
  Chapter 7 - Electrical interfaces
  This chapter focuses on the hardware OpenSequana electrical interfaces.
  Chapter 8 - Power interfaces
  This chapter focuses on the hardware OpenSequana power interfaces.
  Chapter 9 - Firmware interfaces
  This chapter focuses on the firmware features needed in an OpenSequana blade.
- **Chapter 10 - Security**
  This chapter gives recommendations regarding the security of an OpenSequana blade.
- **Chapter 11 - Environmental Requirements**
  This chapter focuses on the environmental requirements mandatory for an OpenSequana blade

In these chapters, we consider the following:

Sentence with “must” means mandatory requirement.

Sentence with “should” means recommended requirement.

Sentence with “may” means nice to have requirement.

2 Acronyms and Abbreviations

Acronym	Meaning
BIOS	Basic Input Output System
BMC	Baseboard Management Controller
BW	Bandwidth
CLI	Command Line Interface
CPU	Central processing unit
CPLD	Complex Programmable Logic Device
DIMM	Dual Inline Memory Module
DLC	Direct Liquid Cooling
DPC	Dynamic Power Control
DWPD	Disk Write Per Day
EMI	Electro Magnetics Interferences
FW	Firmware
FPGA	Field-Programmable Gate Array
FRU	Field replacement unit
GUID	Global Unique Identifier
GPU	Graphical processing unit
HDR	Infiniband at 50 Gbps per lane
HW	Hardware
IPMI	Intelligent Platform Management Interface
KCS	Keyboard Controller Style
MCTP	Management Component Transport Protocol
NIC	Network Interface Controller
NVDIMM	Non-Volatile Dual Inline Memory Module
NVMe	Non-Volatile Memory express disk
OOB	Out Of Band

PXE	Pre-boot eXecution Environment
PCIe	Peripheral Component Interconnect express
PMSM	Platform Management Software Module
PMBUS	Power Management Bus
PUE	Power Usage Effectiveness
ISMA	Islet Monitoring and Administration
RASM	Reliability Availability Serviceability Manageability
RESTFul	REpresentational State Transfer architecture
RPC	Remote Procedure Call
SBI	SideBand interface
SMB	System Management Bus (or SMBus or SMBUS)
SOD	Statement of Direction
SoC	System On Chip
SPOF	Single Point of Failure
SSD	Solid State Disk storage peripheral
SSH	Secure SHell
TBC	To Be Confirmed
TBD	To Be Defined
TCO	Total Cost of Ownership
TDP	Thermal Design Point
TTM	Time To Market
TUE	Total power Usage Effectiveness
U or RU	Rack Unit (=1.75inch or 44.45mm)
UBB	OCP Universal BaseBoard
UEFI	Unified Extensible Firmware Interface

TABLE 1 - ACRONYMS AND ABBREVIATIONS

3 BullSequana XH3000 Overview

3.1 BullSequana XH3000 rack overview

A BullSequanaXH3000 supercomputer is composed of compute racks, each embedding power, cooling, administration capacities for up to 38 OpenSequana Blades. These racks are liquid cooled (except pumps) and all components in the blades must be cooled without air.

Thereby :

- All components in the OpenSequana blades must be cooled without air.

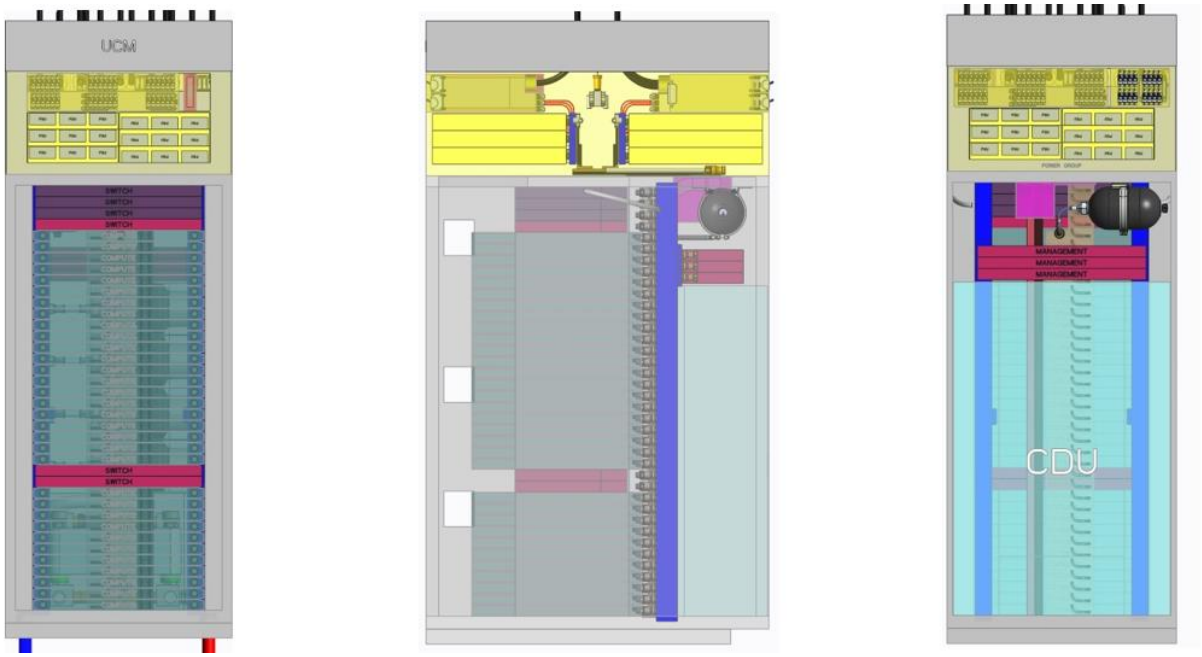


FIGURE 1 - SEQUANA3 RACK

As in any supercomputer, the compute blades are connected to each other within a high bandwidth interconnect.

In order to provide flexibility and reuse standard mezzanines, the connection from blade to the interconnect fabric is done manually from the front of the blade.

- The OpenSequana Blade must connect to external fabric from the front of the blade.

Note that the interconnect switches are OpenSequana blades which comply with this specification.

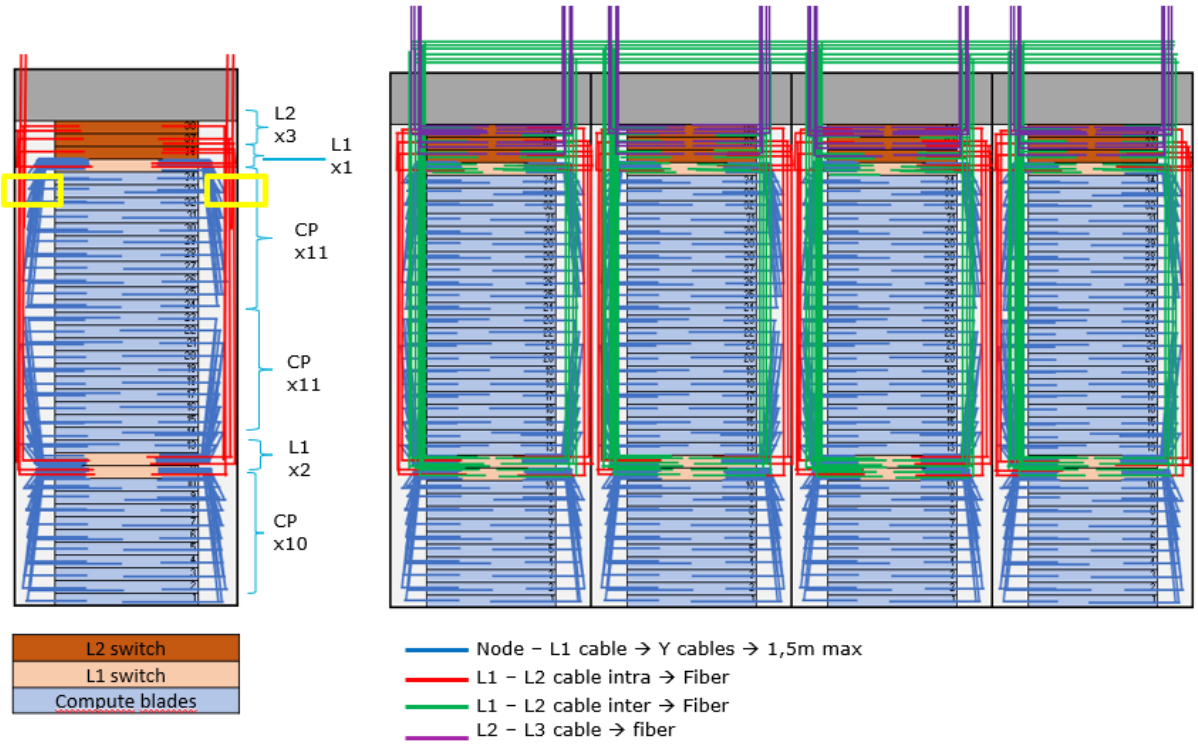


FIGURE 2 - EXAMPLE OF INTERCONNECT CABLING IN SEQUANA3

While the high-speed network is done by front cabling, the rack provides to the blades power, cooling and administration from the rear side of the blade:

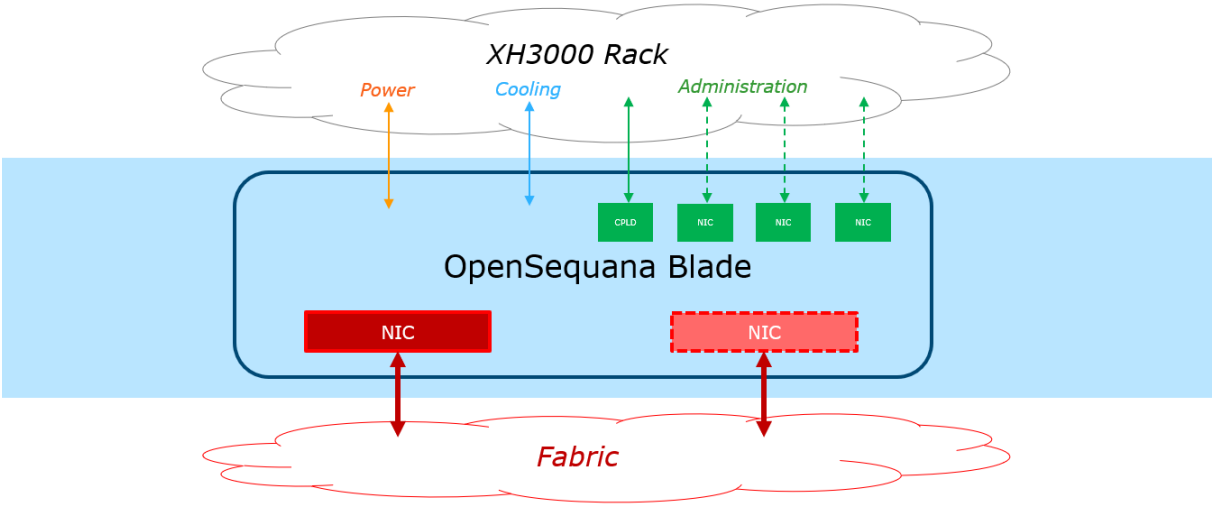


FIGURE 3 -OPENSEQUANA BLADE-RACK INTERFACES

The next figure shows rack services location in the rack:

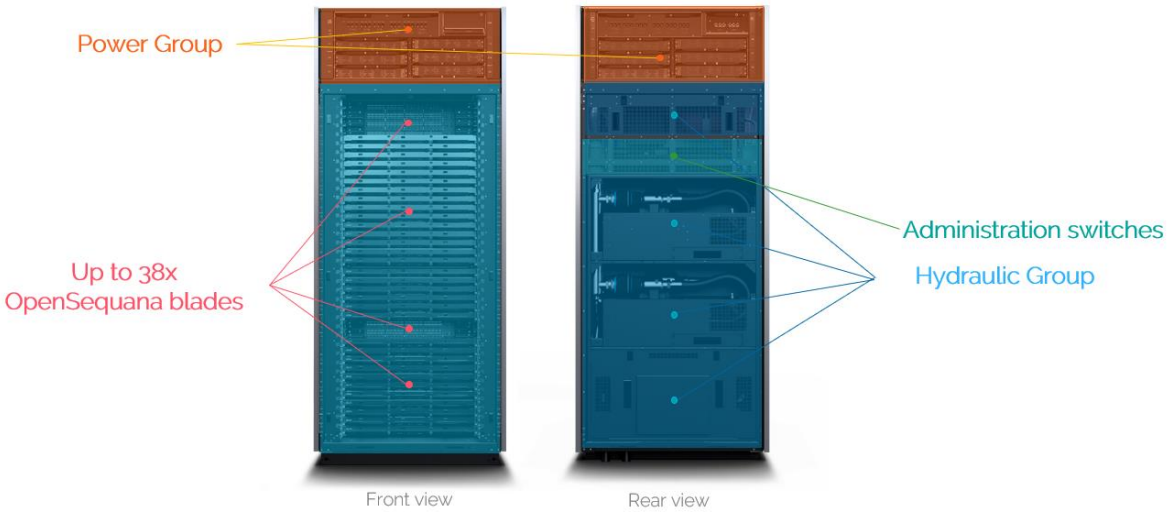


FIGURE 4 - XH3000 DESCRIPTION

The rack infrastructure contains several management controllers :

- One Power Management Controller (PMC) to manage the PSUs in the Power Group
- Two Hydraulic Management Controller (HMC) to manage the two redundant pumps in the Hydraulic Group
- One Rack Management Controller (RMC) to manage the Administration switches and communicates with the other management controllers, including the Baseboard Management Controllers (BMC) in the blades .

3.2 OpenSequana blade functional architecture

A OpenSequana compute blade embeds some compute resources accessible through an operating system. The assembly of the OS and associated hardware (CPU, GPU, NIC...) defines the compute node.

An OpenSequana switch blade embeds one or more interconnect switches (the number of switches is limited by the area needed for the connectors on front side).

The following figure provides examples of OpenSequana blades configuration:

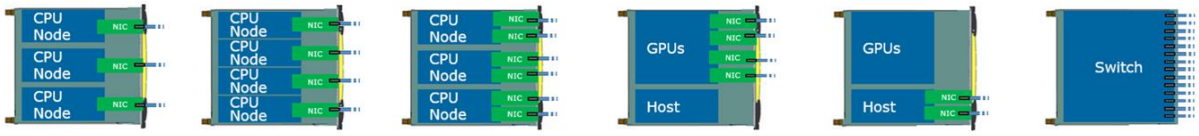


FIGURE 5 - FROM 1 TO 4 MANAGED OR UNMANAGED NODE PER BLADE

In the current implementation, only 3 nodes are managed per blade. Supporting four nodes will require further extensions of the Ethernet network embedded in the rack.

- A OpenSequana compute blade must embed up to 3 nodes.

There are several types of nodes depending on:

- Their nature:
  - General purpose node: usually two CPU sockets with one or two NIC

- GPU node: usually one host (single or bi-socket motherboard) connected to a GPU board extension (UBB for example).
- XPU/APU node : usually several XPU/APU (component providing both CPU and GPU capabilities) interconnected together
- Interconnect node: usually a switch board

- The way they are administrated:
  - Un-managed node
    This type of blade in entirely managed inband from the host for compute node or from the high-speed interconnect for switch blade.
  - Managed node
    This type of node embeds one management controller connected to the administration network of the rack.

- OpenSequana Compute nodes must be managed nodes.

3.3 BullSequana XH3000 rack administration

3.3.1 Rack embedded Ethernet network

By default, all nodes in the blades are connected to embedded switches through 1Gb Ethernet internal links. In the current definition, only 3 nodes are supported per blade. This ethernet link per node is shared between host (main compute element) and Management controller (BMC).

The Ethernet network is sized to address one hundred of compute node (100Gb/s download bandwidth) and provides up to 8 10Gb/s uplinks.

3.3.2 Rack Sideband network

As the administration of all components in the rack is mainly based on Ethernet, there is a need of a simple and reliable network for administration before the management controller is set up or to restart any element in the chain when the Ethernet access is broken. This network is named sideband network.

Sideband Interface (SBI) offers a low-level interface to each blade from the rack manager to:

- Detect the Presence of the blade
  Control the component power (ON/OFF)
- Fix the Management Controller dysfunctions in managed blades
- Discover the component (presence/absence, type of component) to build a rack inventory (based on BoardHwType) and deduce the proper Ethernet interface and dictionary used once the Ethernet is available (Redfish/SNMP/IPMI...)
- Attribute topological addresses to management controllers (see next paragraph)
- Reduce the power consumption of the blade
- Measure the power consumption of unmanaged blades

Sideband interface is based on one RS485 serial link and two discrete signals for presence detection and blade power reduction (see 7.5- Rack Power reduction). The blade is considered slave on the bus (sideband slave), while the sideband master is the rack management controller (RMC). The sideband interface is described in $6.3 and the proper sideband command (frame size and encoding) is detailed in 2.1-1- Sideband interface Specification.

- The OpenSequana blade must be slave on the Sideband bus.

3.3.3 Geographical location in the rack

One Topological Identifier, named TopoId, is introduced to provide to each node in the supercomputer a unique identifier based on the rack number and the location of the node inside the rack. It is used by the BMC to request an IP address to DHCP.

The 38 slots of the racks are allocated with one group number (0 if the blade SBI master is associated to RMC0, 1 if the blade SBI master is associated to RMC1) and one port number per group (from 0 to 19, down to top).

One Sideband Identifier (SBI_ID) is sent by sideband master once the blade is detected. It is defined by 32bits:

- **Reserved[31:28]**
  Provision
- **Rack number Topo[27:16]**
  This is the identifier of the rack, define in the RMC during installation of the rack.
- **Reserved[15:12]**
  Provision
- **Platform type Topo[11:10]**
  0b11 for BullSequana XH3000 platform.
- **Group number Topo[9:8]**
  If RMC0 is the master, Group number is 0'b00
  If RMC1 is the master, Group number is 0'b01
- **Port number Topo[7:0]**
  Define the port number per group

Once received from Sideband Master, the sideband slave keeps SBI_ID readable for the node management controller (BMC). From SBI_ID, BMC can deduce its location within the supercomputer.

The sideband slave must also provide the NodePosition information as the blade can embed up to 3 managed nodes:

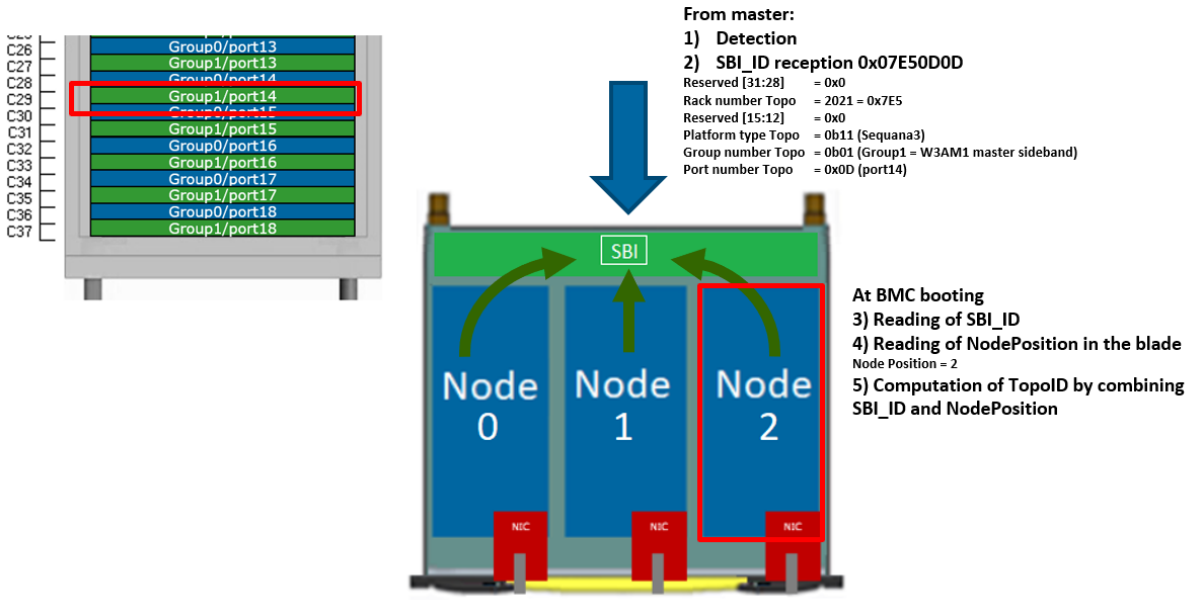


FIGURE 6 - NODEPOSITION INFORMATION

The TopoID is the SBI_ID where the node position has been filled in by the BMC of the node.

- The sideband slave must fill the node position in the TopoId.

4 Mechanical interfaces

This paragraph describes the mechanical interfaces of an OpenSequana blade.

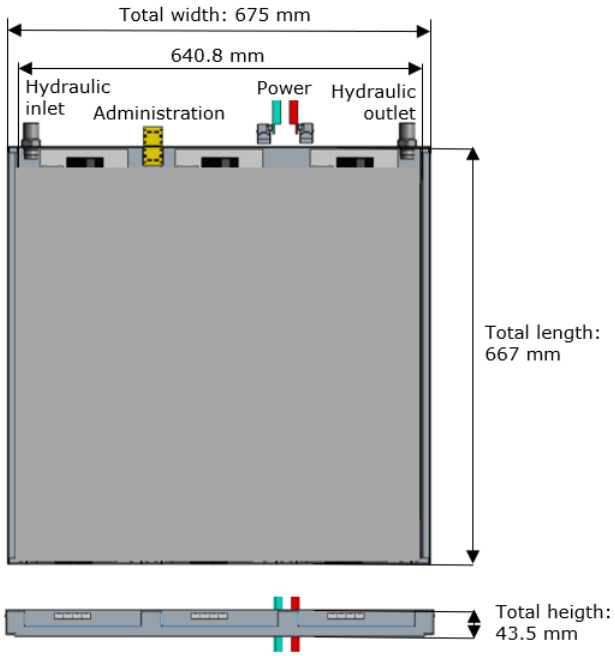
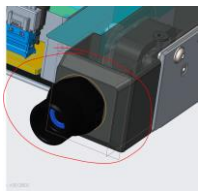


FIGURE 7 - OPENSEQUANA DIMENSION

- The OpenSequana blade mechanical interfacemust implement the proper connector references given in the following table.

Interfaces	Connectors Ref.		More details (pin-out)
Hydraulic connectors	Parker Hannifin NSAC-251-18MM-E		§5 - Thermal interfaces
Administration connector	Amphenol Examax 10128332-12JLF		§6.1- Administration connector pinout

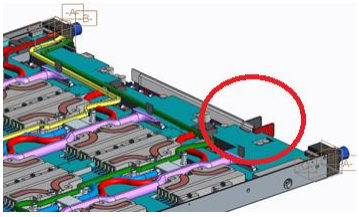
Power connector	Bus bar (ATOS Design)		§7.1- Power Connector

TABLE 2 - CONNECTOR REFERENCES

- These connectors must not be floating, as the floating part of the connection is on the rack side.
- The connectors precise position must be compliant with the below mechanical drawings:

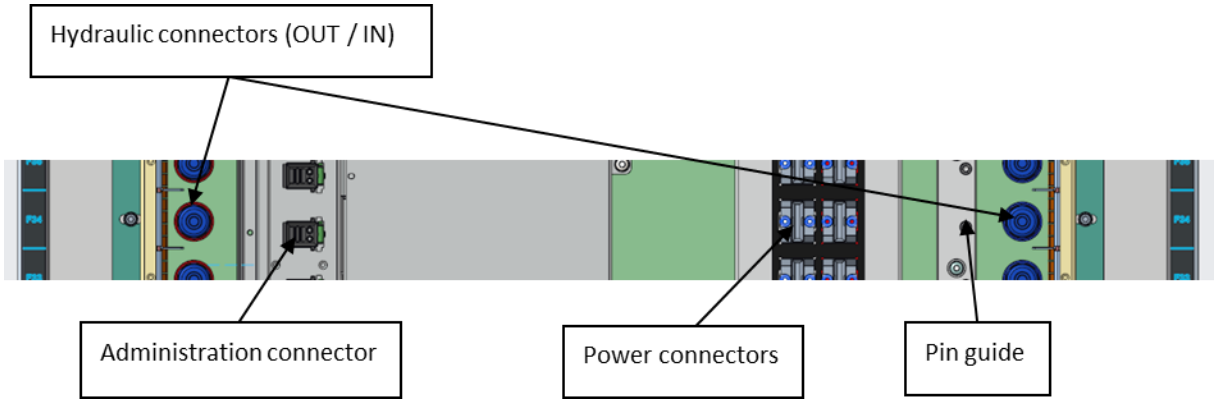


FIGURE 8 - OPENSEQUANA CONNECTORS

In red below is highlighted the mechanical interface

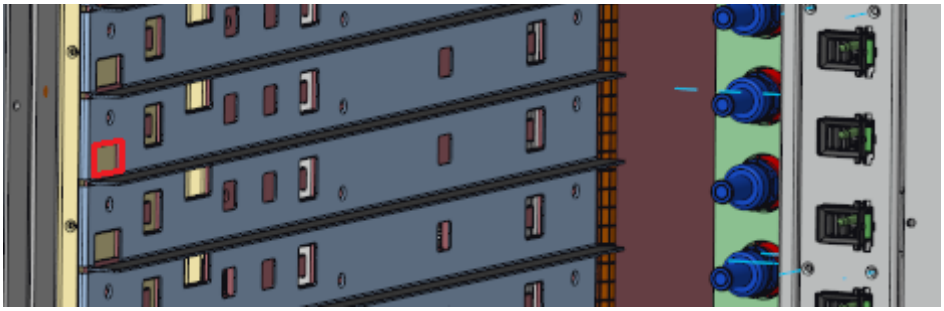


FIGURE 9 - OPENSEQUANA MECHANICAL INTERFACE

Dimensions of the mechanical interfaces:

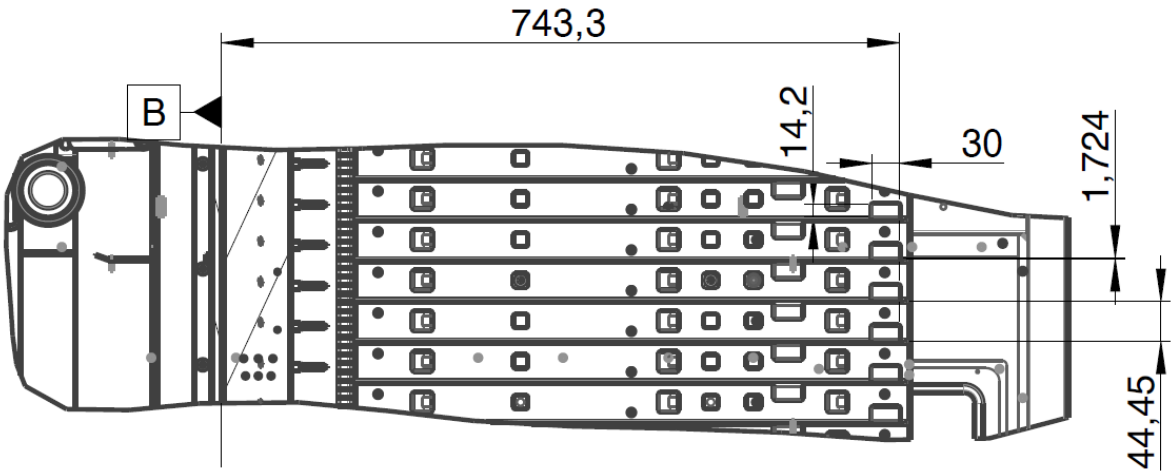


FIGURE 10 - OPENSEQUANA MECHANICAL INTERFACE DIMENSIONS

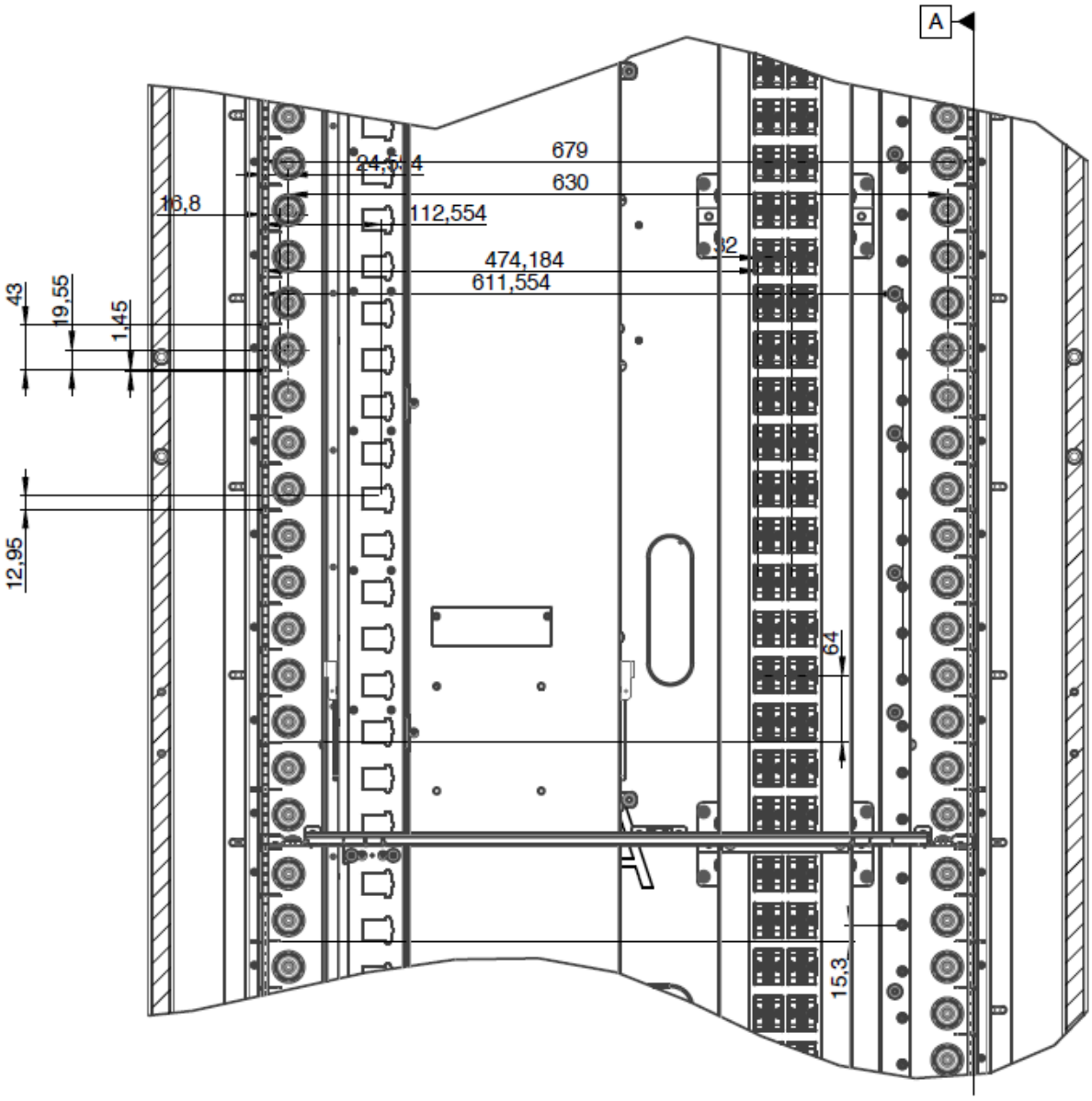


FIGURE 11 - OPENSEQUANA MECHANICAL INTERFACE DIMENSIONS

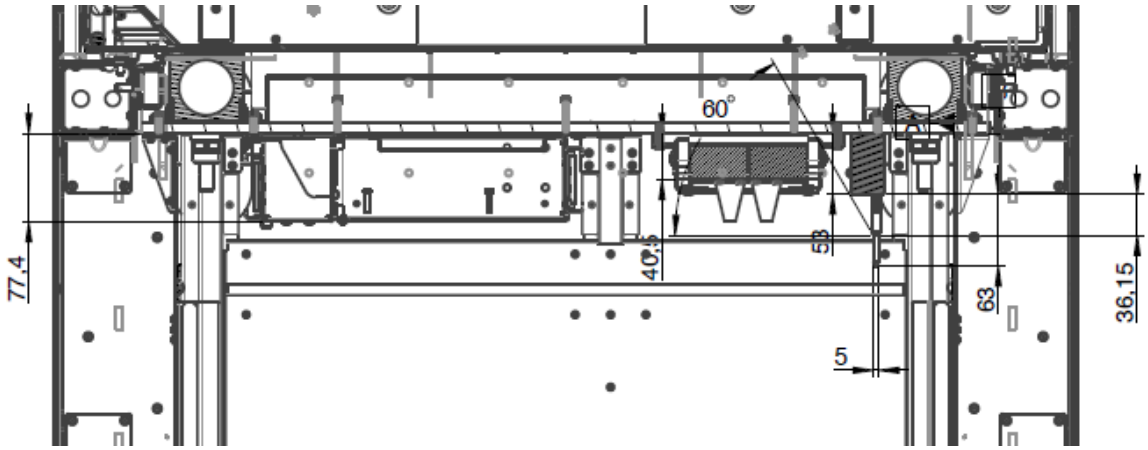


FIGURE 12 - OPENSEQUANA MECHANICAL INTERFACE DIMENSIONS

- A mechanism to ease insertion and lock the blade in the rack must be integrated into the OpenSequana blade.

Rack constraint : Considering hydraulic connectors, power connector and administration connector, insertion effort of the blade in the rack is 240N..

As an example, the following picture shows some existing mechanisms.

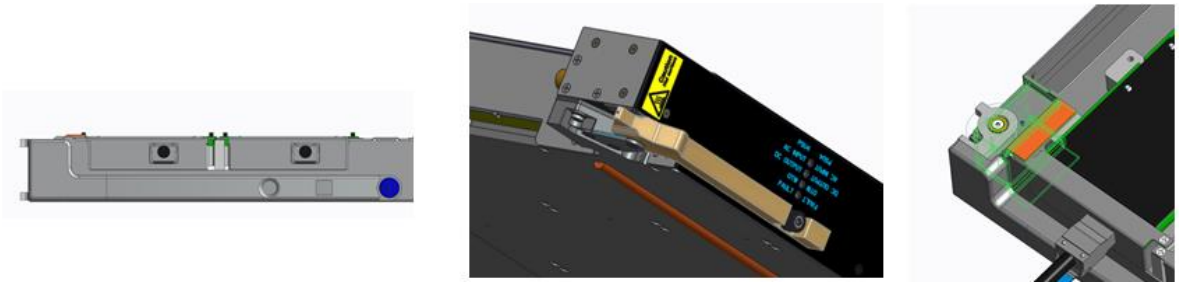


FIGURE 13 - BLADE LOCKING MECHANISM

To ease the development of the product, a 3D step file showing the enclosure and the connectors' position is available on demand :2.1-4- Step file of an OpenSequana blade.

- The OpenSequana blade must weigh less than 35kg to ensure installation in a class5 datacenter floor
- Above 20kg, the maintenance of the OpenSequana blade may need a specific tooling.

5 Thermal interfaces

- The OpenSequana blade must be 100% liquid cooled and be part of the rack hydraulic circuit.

It is forbidden to have any fan in the blade because there is no air path on the rear and the rack can be installed in data center with very low air-conditioning.

- The hydraulic parts in contact with the rack liquid must support MB633 (coolant based on inhibited polyethylene glycol at 33%).

- The hydraulic parts in contact with the rack liquid must be robust to particles smaller than 130µm as the liquid is filtered at 130µm.

- The hydraulic parts in contact with the rack liquid must be in Al6063 T5, without using brazing process to minimize the corrosion risks. FSW process is highly recommended for the aluminium parts.

- The pressure drop between the inlet and outlet of the blade (without the hydraulic couplings) must be 130kPa +/-10% at 6.25L/min with MB633 fluid temperature at 44°C.

The previous pressure drop design target will guarantee the balanced liquid distribution between the different types of blades and the other rack elements (PSU, Administration switch) and then make sure that each compute blade will receive at least 6.25L/min.

- The OpenSequana blade must support the highest possible inlet temperature in the 20°C to 44°C range to maximize free cooling.

The inlet temperature is ensured by the rack Hydraulic Group. This control temperature can be set at the maximum (44°C) or at a lower level (minimum 20°C) if the components within one blade in the rack cannot support this temperature. In case of different blades mixed in the rack, the control temperature is set according to the blade which support the lowest inlet temperature. The highest possible temperature is recommended to optimize Power Usage Effectiveness by using free cooling even in hot days in summer.

- Some local overheating protections must be implemented in the OpenSequana blade.

- Some sensors must be placed in relevant areas and thresholds must be computed to allow two levels of overheating alarm:

- Warning : once the Warning threshold reach, an event must be sent to the RMC and remote RSyslog

- Critical: one the Critical threshold is reached, an event must be sent to the RMC and remote RSyslog, then the blade must suicide itself (power off of the host and of the management part).

6 Electrical interfaces

6.1 Administration connector pinout

The administration connector provides the ethernet and sideband links to the blade.

- All signals in administration connector must support the hotplug of the OpenSequana blade.

- The administration connector must implement the pinout and signal characteristics described in the next figures and table.

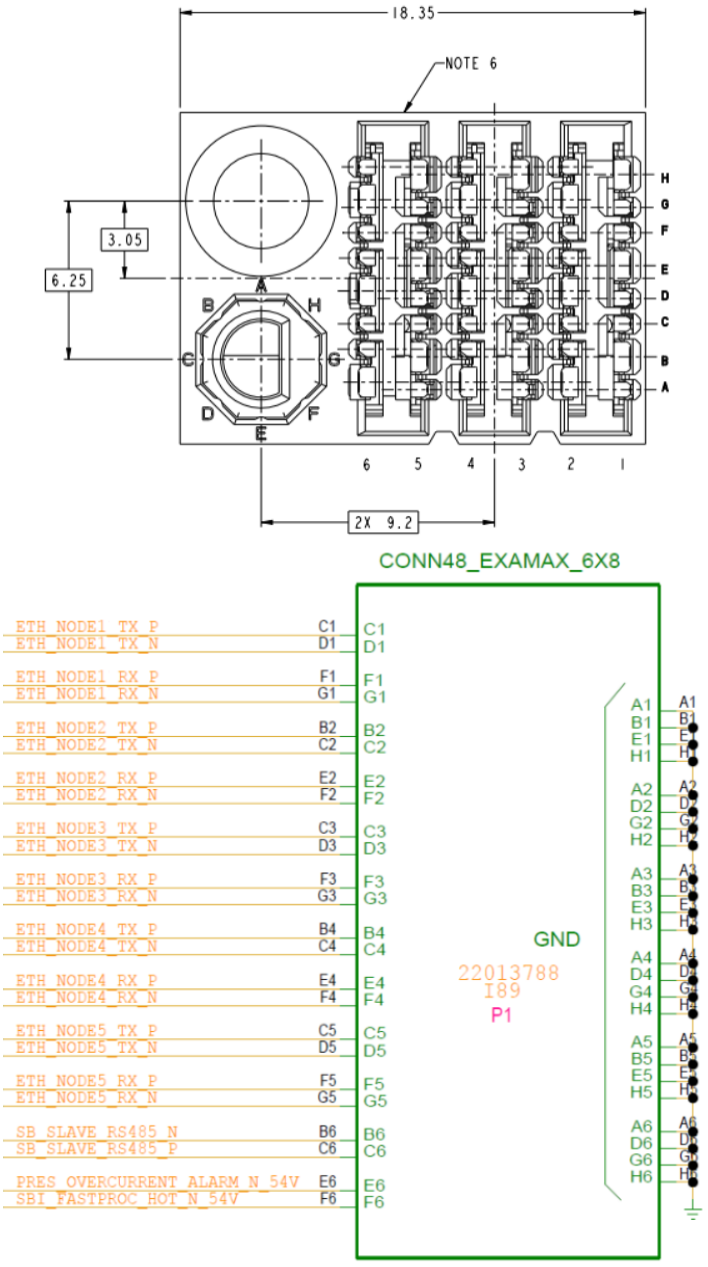


FIGURE 14 - ADMINISTRATION CONNECTOR OF THE BLADE

Pin	Signal Name	Input/Ouput (from the blade)	Details
A1	GND		
A2	GND		
A3	GND		
A4	GND		
A5	GND		
A6	GND		
B1	GND		
B2	ETH_NODE2_TX_P	O	Ethernet 1000 base X to manage node2
B3	GND		
B4	GND		
B5	GND		
B6	SB_SLAVE_RS485_N	I/O	Sideband interface RS485
C1	ETH_NODE1_TX_P	O	Ethernet 1000 base X to manage the node1
C2	ETH_NODE2_TX_N	O	Ethernet 1000 base X to manage the node2
C3	ETH_NODE3_TX_P	O	Ethernet 1000 base X to manage the node3
C4	GND		
C5	GND		
C6	SB_SLAVE_RS485_P	I/O	Sideband interface RS485
D1	ETH_NODE1_TX_N	O	Ethernet 1000 base X to manage the node1
D2	GND		
D3	ETH_NODE3_TX_N	O	Ethernet 1000 base X to manage the node3
D4	GND		
D5	GND		
D6	GND		
E1	GND		
E2	ETH_NODE2_RX_P	I	Ethernet 1000 base X to manage the node2

E3	GND		
E4	GND		
E5	GND		
E6	PRES_OVERCUR-RENT_ALARM_N_54V	O	Sideband Presence Discrete: Blade present : 0VDC Blade raising interrupt : 0 to 54VDC 10kHz square signal: Blade not present : tight to 54VDC by the master
F1	ETH_NODE1_RX_P	I	Ethernet 1000 base X to manage the node1
F2	ETH_NODE2_RX_N	I	Ethernet 1000 base X to manage the node2
F3	ETH_NODE3_RX_P	I	Ethernet 1000 base X to manage the node3
F4	GND		
F5	GND		
F6	SBI_FASTPROC_HOT_N_54V	I	Sideband Fastprochot Discrete: **0VDC** : rack requests the blade to throttle **54VDC** : normal state
G1	ETH_NODE1_RX_N	I	Ethernet 1000 base X to manage the node1
G2	GND		
G3	ETH_NODE3_RX_N	I	Ethernet 1000 base X to manage the node3
G4	GND		
G5	GND		
G6	GND		
H1	GND		
H2	GND		
H3	GND		
H4	GND		
H5	GND		
H6	GND		

TABLE 3 - SIGNALS NAME IN THE ADMINISTRATION CONNECTOR

6.2 Ethernet budget

This paragraph focuses only on the Ethernet 1000 base X hardware interface.

The functional interface is described in 3.3.1- Rack embedded Ethernet network

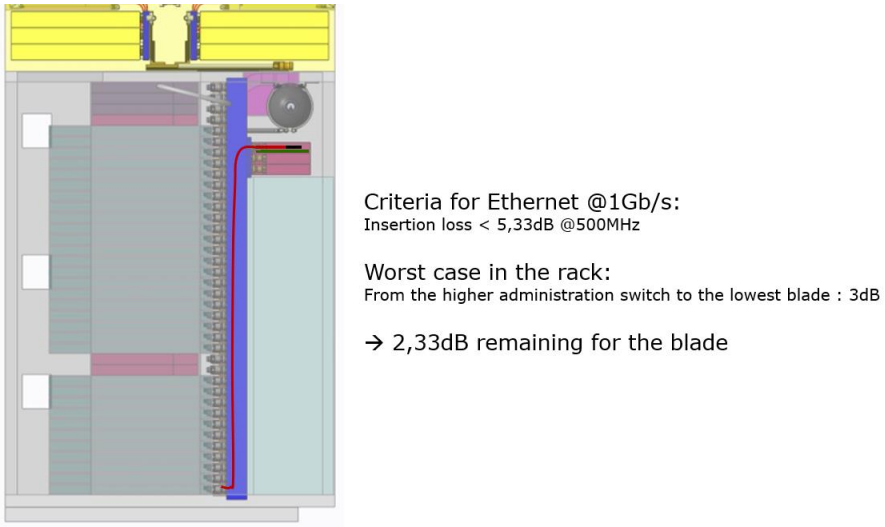


FIGURE 15 - ETHERNET BUDGET OVERALL

- The Ethernet budget in the OpenSequana blade (Examax connector included) must be lower than 2.33dB @500MHz.

- If needed, an Ethernet retimer may be added in the OpenSequana blade.

As this budget is challenging, an Ethernet retimer is placed right after the blade connector in some OpenSequana blades.

6.3 Sideband interface description

6.3.1 Overview

The sideband interface is used for low level management of the nodes (see 3.3.2- Rack Sideband network). The interface is based on a RS485 half-duplex link and 2 discrete IOs. The protocol is master/slave based. The rack management boards are the two masters, and the nodes are the slaves.

The master always starts the communication and wait for the slave answer.

These signals are used as follow:

- The RS485 is used to dispatch the topology of the rack to the nodes. and to perform blade remote control such as powering ON/OFF the node, force the BMC to boot on backup memory…

- The discrete IOs are used to transmit information as fast as possible:

  - First IO named SBI_FAST_PROC_HOT_N_54V is from master to slave and used to throttle the node

  - Second IO named SBI_OVERCURRENT_ALARM_N_54V is from slave to master. and used for slave presence detection and interrupt request.

6.3.2 IO control

The first signal SBI_FAST_PROC_HOT_N_54V is asserted to 0 by the master when the slave board must throttle to limit the power consumption.

The second signal is a presence and interrupt signal sent from the slave to the master. The signal is asserted to 0 when the slave is present, an interrupt signal is a square signal at 10kHz. When the slave is not present, the signal is at 1.

- The OpenSequana blade must assert the presence and alarm signal (SBI_OVERCURRENT_ALARM_N_54V) to 0 or issue a square signal at 10kHz to raise an interrupt.

- The OpenSequana blade must throttle to limit power consumption when it received the sideband discrete SBI_FAST_PROC_HOT_N_54V signal asserted to 0 (0V-54V).

- The OpenSequana blade must implement the presence signal and the throttle signal according to the following schematics.

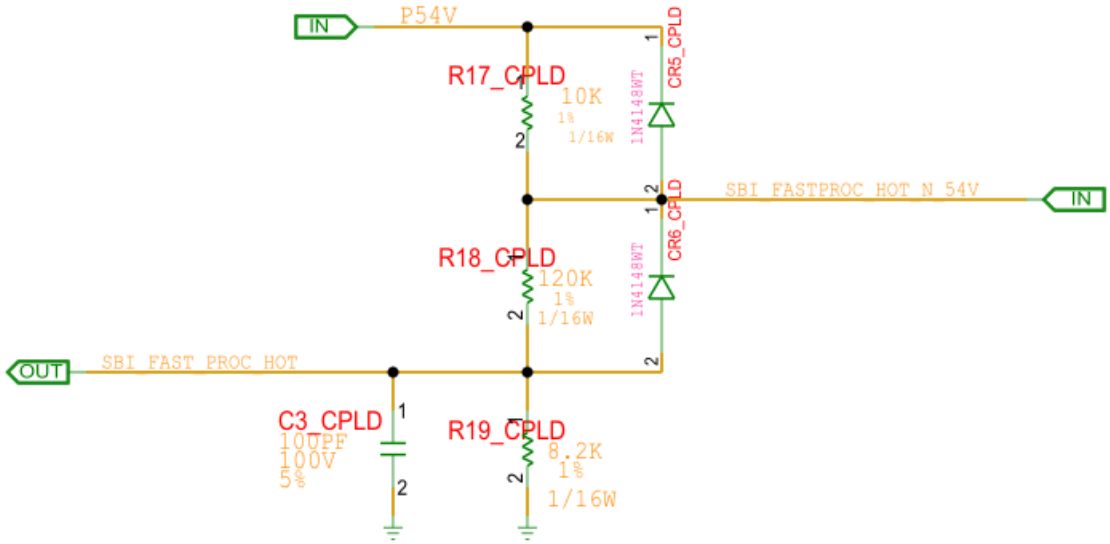


FIGURE 16 - SIDEBAND SLAVE SBI_FASTPROC_HOT_N_54V SCHEMATIC

FIGURE 17 - SIDEBAND SLAVE PRES_OVERCURRENT_ALARM_N_54V SCHEMATIC

6.3.3 RS485 link

This paragraph focuses only on the hardware interface. The functional interface is described in §8-Firmware interfaces.

The Sideband interface is a half-duplex RS485 serial link between a sideband master implemented in administration board of the rack and sideband slave implemented inside the blade.

- The blade must implement a sideband interface through a half-duplex RS485 link configured at 250kbaud.

There is no insertion loss constraint on this RS485 differential pair because it is routed in the same rack cable octopus than the Ethernet_1G and is far more noise immunized.

A sideband exchange always starts by a request from the master.

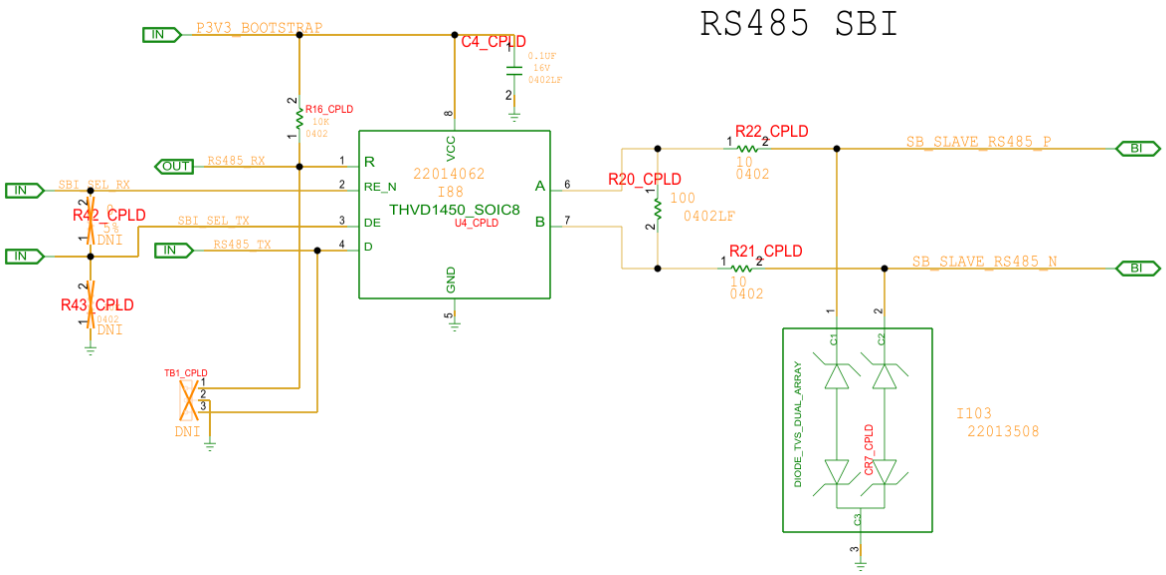


FIGURE 18 - SIDEBAND SLAVE RS85 TRANSCEIVER SCHEMATIC

6.4 Sideband protocol

RS485 sideband communication between sideband master and slaves is based on the principle of request and response.

The master initiates the communication by sending a command request. The slave then responds by an acknowledge and potentially with data.

- The data read and write to the slave must be manage as a 256 bytes memory.

- The first 128bytes must be Read-Only.

- The bytes from index 128 to 255 must be Read-Write accessible.

- The blade sideband slave must respond to master request with an acknowledge and data if needed, using the protocol described.

Below a representation of a typical sideband frame.

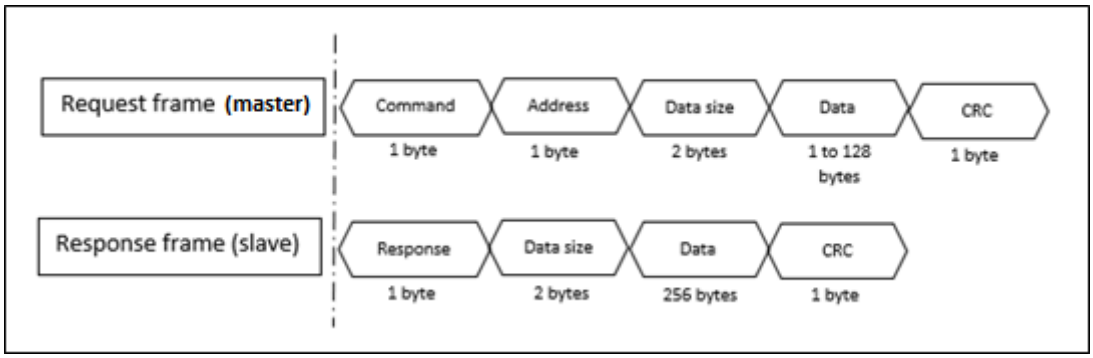


FIGURE 19 - TYPICAL SIDEBAND FRAME

Each byte is sent via Tx and received via Rx bit per bit starting with MSB.

- The blade must interpret the master request frame and generate the response frame according to the protocol described in document BNT_CARTES_4721_Sideband_Registers_specification.

This document provides the two commands used for read and write as describe in the following table.

Command name	Command value	Description
Status refresh	0xC3	The status refresh command requires the slave to return all the 256 bytes current value of the side-band memory
Config refresh	0xC4	The config refresh command sends 128 bytes to write to the sideband memory at index 128. The slave responds with the 256 bytes (containing the 128 bytes wrote) of the sideband memory.

TABLE 4 - SIDEBAND READ AND WRITE COMMANDS

7 Power interfaces

7.1 Power Connector

The mating connector of the power interface in the rack is Teico C-1643903 :

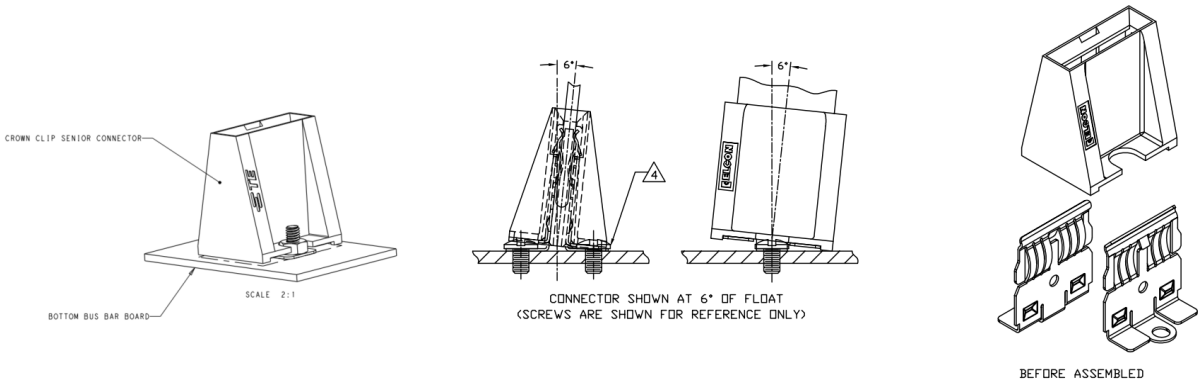


FIGURE 20 - BULLSEQUANAXH3000 POWER CONNECTOR

- The OpenSequana blade must have a plate design following Teico C-1643903 recommendation except for the power connector height which must be 22.00 +/- 0.3 mm.

This height differs from the 25.40 +/-0.38 mm recommended by Teico due to dilatation phenomena in the rack

- The OpenSequana blade copper plate must comply with the Teico "Gold plated" rule for the surface treatment.

Indeed BullSequana XH3000 rack infrastructure is not compatible with copper alloy, silver plated of nickel plated.

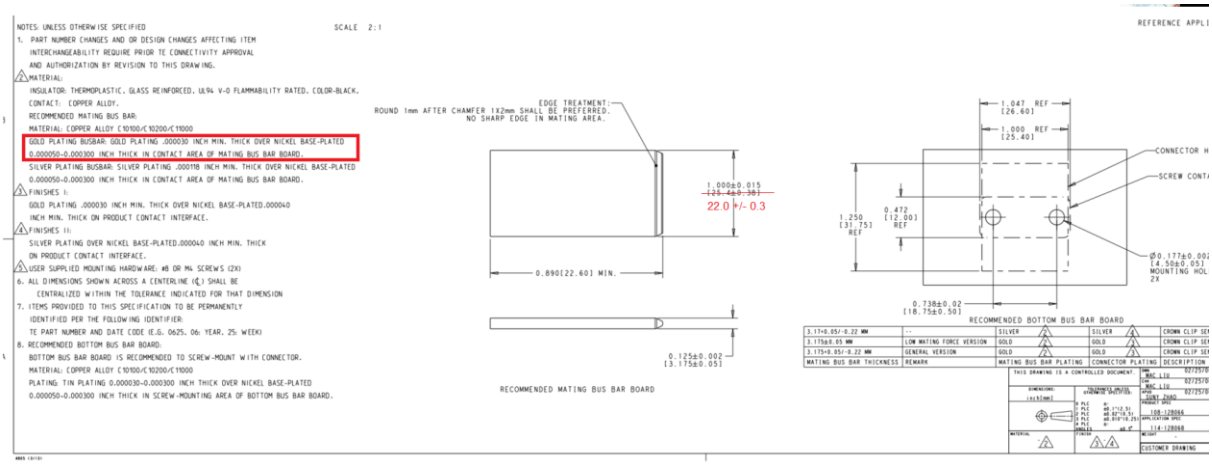


FIGURE 21 - ONLY CHANGE IN TEICO MATING RECOMMENDATION

7.2 Power interface

The power interface consists of a bus bar bringing 55VDC to the blade.

- OpenSequana blade must be hot-swappable in a running rack.

- Surprise hot-swap of a running blade must not cause any physical damage.
- The OpenSequana blade must embed a circuitry to avoid over-current in case of hotswap.

- The OpenSequana blade must embed a fuse to protect the material and the users in the case of malfunctioning situation.

- The fuse maximum interrupting current must be at least 10kA

- The OpenSequana blade must support a nominal voltage of 55V for normal operation, with max 5.5V peak-peak variation.

However, when this is power outage, the rack power could be secured by an optional ultracapacitor module and in this case the nominal voltage is 51V when the ultracapacitor are active.

- The OpenSequana blade must support a nominal voltage of 51V, with a maximum 5.5V peak-peak variation, during the 2s this power backup mechanism may last.

- The OpenSequana blade must consume maximum 6000W including peak.

Going above this 6000W limit is not compatible with the rack power connector and would create severe local increases of temperature on the bus bar.

7.3 Rack constraints for Steady Power budget

BullSequana XH3000 provides 147kW and is able to evacuate it in 40°C data center water.

Having 38 identical blades leads to a steady power budget of 3868W per blade. However, most of the time, the rack embeds a combination and compute blades and switches. Considering a configuration with 4x 750W switches and 32 compute blades, the steady power budget per compute blade is 4500W.

Most of the OpenSequana compute blades are between 4000W and 4500W because it offers a good compromise between cost (having embedded switches limits the use of expensive optical cables) and high-water temperature.

However, the rack can accept up to 6000W per blade, but the number of blades supported per rack is smaller. For example, the OpenSequana rack supports maximum 24x 6000W blades plus 4x 750W switches or 6x 500W switches.

- The maximum number of OpenSequana blades per OpenSequana rack must be defined to ensure the proper maximum steady budget per blade.

7.4 Rack constraints for Peak Power budget

CPU and GPU tend to have, in addition of the steady power (TDP), some peak power (turbo mode, overcurrent going for Idle to compute). For CPU, these peaks last usually less than 5ms with a frequency of 10Hz (around 10 peaks per second). For GPU, peak duration can be up to 50ms. It has no impact on the thermal behaviour of the blade but it is visible to power distribution and PSUs inside the rack.

Some GPU and CPU could have very high peak during short duration (for example 400µs). In this case, local capacitor should be added on the blade to provide this very short peak power and ensure that the power visible at blade power connector is not exceeding 6000W.

- The OpenSequana blade may implement local capacitor to avoid exceeding 6000W with components with high peak power in very short duration.

These peaks must be considered in the blade design, as well as simulated at the rack level. The major risk is to have these peaks synchronized and creating a resonance.

- The OpenSequana blade design must avoid that a synchronized load could bring the 55V at the rack above 5.5V peak-to-peak.

The PSU on the rack can deliver more power for short duration as on the table below.

	PSU Power	Rack Power (35 PSU)	Margin
Nominal power	4200	147000	
Max continuous	4500	157500	107%
Max peak 100ms	4860	170100	116%
Max peak 10ms	5034	176190	120%
Max peak 5ms	5207	182245	124%
Max peak 1-2ms	5480	191800	130%

TABLE 5 - PEAK POWER BUDGET

There are 4 tri-phase power lines for 35 PSU (+ 1 not taken in account because reserved for redundancy) with efficiency around 95% and power factor at 99%. The current on these tri-phase power lines must never exceed the rated current of 63A even during short peak. This could be the limiting factor to allow some power peak.

A LT Spice model of the rack is available and could be shared to perform this simulation 2.1-5-LTSpice model of the rack. It encompasses power supply model of the rack (characteristics, output capacitance) as well as distribution to the compute blades (inductance, capacitance).

- The maximum number of OpenSequana blades per OpenSequana rack must be defined to never exceed the rated current of 63A even during short peak .

7.5 Rack Power reduction

The OpenSequana rack implements some mechanism to avoid stopping the rack if there is an overpower consumption. If the PSU detects an overpower consumption, the rack propagates the discrete sideband signal `SBI_FASTPROC_HOT_N_54V` to every blade. This signal is also asserted when some ultracapacitors provide the power during an AC power outage, to extend the duration of the power outage supported.

- The OpenSequana blade must decrease its power consumption by at least 50% in less than 100 µs, on reception of `SBI_FASTPROC_HOT_N_54V`.

Depending on the compute components, it could be done by using therm_trip interface, Prochot or power brake interface on CPU/GPU.

8 Firmware interfaces

8.1 Ethernet Interface

This paragraph applies to nodes with BMC for out-of-band management.

- Node must be administrated through 1Gb/s Ethernet.

- Node Ethernet management access must be shared between BMC and Host.

-BMC should be available when host is unpowered.

- The blade must compute TopoID by combining SBI_ID and NodePosition

- BMC should provide a command to get TopoID.

An external DHCP server is responsible for IP attribution to BMC and host (OS in compute node).

In OpenSequana, this IP is not allocated based on MAC address because the IP address must be unchanged when a blade is replaced. To identify BMC from other IP equipments and ensure unicity and persistence of IP, BMC DHCP request should use the option60 parameter (VendorClassIdentifier) and the option61 parameter (ClientID).

VendorClassIdentifier must be

- vendor name : TBC (BULL for blades developed internally)
- PlatformName : SQ3
- Version_ID : x (0 for the first version)

ClientID must be contain the vendor type, Enterprise number and the topoID of the node (SBI_ID + node position as described in 3.3.3- Geographical location in the rack .

Other components, such as OS in the compute node, service node can also get an IP on the administration ethernet network. The common way is through option82, but it could be different.

- To get a global IP, BMC must send DHCP request filled with MAC address, option60 and option61 parameters.

- To get a local IP to support rack standalone network capability, BMC FW should support Zeroconf.

8.2 Redfish interface

Redfish interface is the recommended interface for a managed node.

Redfish interface allows definition of roles with management of permission regarding the defined roles. Then, by creating a user account, association is made between the account (user) and a role (permissions).

Three roles are defined with the following permissions:

- **Administrator**
  An administrator role has the following permission
  - Authentication (`Login`)
  - Creation of user account (`ConfigureUsers`)
  - Activation/Desactivation of SSH interface (`ConfigureManager`)
  - Configuration of the node (`ConfigureSelf/ConfigureComponents`)
  - Reading of sensors/variable

- **Operator**
  A support role has the following permission
  - Authentication (`Login`)
  - Configuration of the node (`ConfigureSelf/ConfigureComponents`)
  - Reading of sensors/variable
- **Read-only**
  A monitor role has the following permission
  - Authentication (`Login`)
  - Reading of sensors/variable

At the creation of the user account, a role is attributed to the user.

The following table provides Redfish API reference for OpenSequana compute nodes as implemented in BMC firmware.

This table describes the Redfish High-level data model, where each data model object relies on the schema (json file) provided by Redfish Standard. We assume that the version of bundle of Redfish schema files supported in BMC is v2020.1 or later.

Redfish Resource		Redfish URI (Uniform Resource Identifier)	Mapping to OpenSequana concepts
Service root		/redfish/v1	
Chassis collection		/redfish/v1/Chassis	Chassis collection contains Redfish Chassis instances corresponding to the managed compute node board and mezzanines
	Chassis	/redfish/v1/Chassis/{ChassisID}	ChassisID can be any board in the BMC's node (`CPU_Board`, `Mezzanine ...`) or represent the full blade (`Blade`)
	Sensor	/redfish/v1/Chassis/{ChassisID}/Sensors	List of thermal and power sensors.
	Power	/redfish/v1/Chassis/{ChassisID}/Power	Service Providing power metrics capabilities.
	Thermal	/redfish/v1/Chassis/{ChassisID}/Thermal	Service Providing thermal metrics capabilities.
System collection		/redfish/v1/Systems	System collection contains one instance representing the BMC's node and its properties (CPU, BIOS FW, memory)
	System	/redfish/v1/Systems/{SystemID}	SystemID is: **`Host`**
Manager collection		/redfish/v1/Managers	Manager collection contains one manager instance representing the BMC managing the compute node

	Manager	/redfish/v1/Managers/{ManagerID}	In case of managed object, managerID is: **BMC**
Update Service		/redfish/v1/UpdateService	UpdateService provides Firmware update capabilities and actions
Session Service		/redfish/v1/SessionService	SessionService describes the properties related to user connections to the Redfish service
Task Service		/redfish/v1/TaskService	TaskService is used for the management of long-duration operations (eg. FW Update)
Event Service		/redfish/v1/EventService	EventService is used to manage user subscription to BMC events

TABLE 6 - REDFISH API REFERENCE

The following Redfish fields (attributes) are not part of standard Redfish interface.

Under a Chassis element, the following OEM fields are added:

- **`BoardHwType`**
  Give the hardware type of the board.
- **`BoardRevId`**
  Give the revision version of the board. It starts at 0 and is increased each time an hardware modification has been done on the board.
- **`TopoId`**
  Give the SBI topoId of the node.

- Out-of-band management interface (BMC) must comply with Redfish API v1.1.0

- Out-of-band management interface (BMC) must support Redfish schema bundle version 2020.1.

Higher version of Redfish schema (2020.3) could be supported later according to OpenBMC stack evolutions.

8.3 Other interfaces

- BMC should log its notifications to an external Rsyslog hosted by an external administration server.

- BMC should implement a NTP client.

8.4 Administration

8.4.1 Initialization

Note that the term “start” means that the corresponding logic is powered-on and “stop” that it is powered-off.

- The blade management components must start automatically when 55V is available

- BMC should provide a command to power-on the compute node

- The initialization of the blade must follow the steps described in the following table, either at blade insertion or at rack booting:

Steps	
0. Either the blade is not in the rack or the rack is not powered	
1. 55VDC is available (Rack start or blade insertion)	
2. SBI slave starts	
3a. SBI slave signals its presence through side-band to master.	3b. SBI slave automatically powers up the BMC
4a. Master sends TopoID to SBI Slave	4b. BMC starts
5. SBI Slave receives SBI_ID	...
	6. BMC reads SBI_ID and generate node position information to compute TopoID, ClientID and LocalID
	7. BMC sends DHCP request with option60 and option61 parameters
	8. BMC Redfish interface is now available locally from RMC (for automatic inventory build)
9. BMC puts BMC_State discrete to "BMC_OK" in the SBI register	
	10. Through BMC Redfish, a user can turn on the host (OS)
	11. The OS boots according to BIOS configuration.

TABLE 7 - INITIALIZATION SEQUENCE

8.4.2 Stop

- The blade must power-off the node (Host + BMC) on SBI request.

The usual steps to stop an OpenSequana blade are:

1. From BMC, shut down gracefully the host's OS
2. From BMC, power off the host
3. From BMC, shut down gracefully the BMC
4. From SBI, power off the node

8.4.3 Inventory

At the rack level, The blade must provide some inventory functions. For auto-discovery of the rack (rack start or blade insertion):

- The blade must provide the node type on SBI request.

- Node type must be composed of Board ID (8bits) + Board Revision (3bits).

- Node type must be accessible even if the motherboard is not powered.

- BMC must provide a command to read the node type.

Static information such as manufacturer, product ID, or serial number (usually stored in FRU chips following IPMI Platform Management structures) should be made available to RMC and management infrastructure through Redfish.

- BMC should expose static board information (node type, manufacturer, serial number...) in a Redfish Assembly schema under the Redfish Chassis corresponding to each board.

- BMC should expose static blade information (node type, manufacturer, serial number...) in a Redfish Assembly schema under the Redfish Chassis corresponding to the blade.

- BMC should provide a command to get information of main components.

- BMC should provide a command to display sensor list of the node.

- BMC should provide a command to retrieve sensors information.

- BMC must provide commands to retrieve version of all firmwares.

8.4.4 Health monitoring

Power Monitoring

- The blade must return power state of a BMC (BMC_PowerGood) on SBI request.

- The blade should provide a Redfish command to return Linux state of a BMC (BMC_State=BMC_OK).

- BMC should provide a command to get the power status of the node.

- BMC should monitor voltage sensors.

- BMC should monitor power-consumption sensors.

- BMC should provide a command to display power sensors (voltage, current and power-consumption).

Temperature Monitoring

- BMC should monitor 55VDC hotswap temperature on power board.

- BMC should monitor of main components.

- BMC must provide a command to display temperature sensor value for main components.

Error Monitoring

- All errors in the node should be logged and reported.

Eventing

An event is a detectable occurrence that has significance for platform management (system status change, failures, etc.). Events are not necessarily conducting to a notification (Alert). However, they must be locally logged and send to the remote RSyslog.

Events are categorized following their significance:

- Informational: the event does not require any immediate action and does not represent a failure. Usually, this event doesn't lead to a notification toward administration tools,
- Warning: the event is generated when a sensor reaches a predefined threshold. Warning event usually lead to a notification toward administration tools to prompt system operator to take the necessary action to prevent a failure occurring,
- Critical/Fatal (Failure): the event is generated when a service is operating below the normal predefined parameters or thresholds. System is impacted by performance degradation or loss of functionality. A recovering action should be performed. This kind of event in necessarily followed by high-priority notification toward administration tools.

- BMC should log any event in a Journal (called Event log).

- BMC should send any event to external RSyslog.

- BMC should provide a command to read Event log.

Regarding power, error logging is only possible on warning threshold for some voltage/current measured, since critical threshold will trig a stop of the node (however, such event is logged into Event Log).

Alerting

An alert is a notification that a particular event (or series of events) has occurred.
Alerting mechanism is usually based on client subscription and server notification methods, such as http notification.

- BMC must provide alerting mechanism via Rsyslog

- BMC must provide a command to subscribe to an alert.

- BMC must provide a command to unsubscribe to an alert.

- Warning temperatures must generate an alert.

- Critical temperature must generate an alert.

- Warning voltage must generate an alert.

Regarding power, alerting is only possible on warning threshold for some voltage/current measured, since critical threshold will trig a stop of the node (however, such event is logged into Event Log).

- CPU throttling must generate an alert.

8.4.5 Power measurement

Power consumption

- BMC must provide a command to get power consumption for the whole blade node.

- Compute node input power consumption must be measured with 5% accuracy on 55VDC (from 20% to 100% of PMax at nominal liquid temperature).

- OS must provide a command to get power consumption for the all computing elements.

Power accounting

- BMC should implement a energy accounting mechanism to measure consumed energy by the node from its start.

- BMC should provide a command to get the power accounting value (energy in kJ and time stamp in second).

Proposed Redfish command response format (json):

Variable	Description
Power_data_collection_status	Data structure containing BMC register statuses (see below)
PWR_Register_Status	BMC power register status (0 if data not available, 1 if data available)
SPL_Register_Status	BMC sample register status (0 if data not available, 1 if data available)
PWR_Overflow_Status	BMC power data register overflow status (0 if free space is still available, 1 if overflow)
SPL_Overflow_Status	BMC sample data register overflow status (0 if free space is still available, 1 if overflow)

Reading_Timestamp	BMC time corresponding to register reading time
Energy	Energy value in Joules
Samples	Number of samples used to compute the energy
Instant_Power	Instant power value in milliwatt

TABLE 8 - REDFISH COMMAND RESPONSE FORMAT

8.4.6 Setting

- BMC should provide a Redfish command to change BIOS setting.

8.5 Maintainability

8.5.1 FW Management

- Slave SBI must be upgradable through SBI or BMC Redfish command.

- BMC should provide a commands to upgrade any Firmware image in the node

8.5.2 Diagnostic

- The OpenSequana blade should provide LEDs on front face for BMC Ethernet (Node status LED, Ethernet LED, blue identification LED)

- BMC should provide a command to manage the blue Identification LED

8.5.3 Recovery Operations

- BMC should provide Redfish commands to repair host OS (reboot, reset, restore BIOS parameters to default ...)

- The blade should clear CMOS one node on SBI request

- BMC should provide Redfish commands to repair itself (Warm reset, Cold reset...).

- The blade should reset a node power supply (Node power cycle or Cold reset) on SBI request.

- The blade should restore-to-default one node on SBI request.

8.5.4 Safety

Safety prevents from placing a technician or material in danger.

- The blade design must comply with EN62368-1
- The blade design must comply with Directive 2014/35/EU (Low Voltage Directive).
- The blade must not have any voltage above 75VDC.
- PCB inflammability class must be UL V-0.

- Ground connection to side plane must mate first regarding other signals during blade insertion.
- Last mated contact when blade extraction must be Ground connection.
- An extracted blade must not have any energy stored except RTC battery.
- The blade insertion must not induce any hydraulic projection/drop.
- The blade extraction must not induce any hydraulic projection/drop.

9 Security

This paragraph states some recommendation regarding security of the OpenSequana blade. Those recommendations are followed for ATOS blade.

- No SSH
  SSH interface should not be available once the machine is deployed. For very few and determined occasions (factory setting), it could be used. However, it is ATOS recommendation to disactivate it once the blade is delivered to a customer.
- Secure boot based on a Hardware root of trust

Secured boot is ensured by a chain of trust. In a chain of trust, each element verifies the integrity of the next element before launching it. The usual mechanism is based on encrypted signature.

After firmware compilation, a signature is created with the content of the image. Then the signature is encrypted with a private key. The result called "hash", is added to the firmware image. The corresponding public key is furnished to the element responsible for checking the Fw_image integrity (usually the element before in a chain of trust).

During boot, each element verifies the integrity of the next element before launching it. The following sketch illustrates the principle with only two linked elements of the chain during boot:

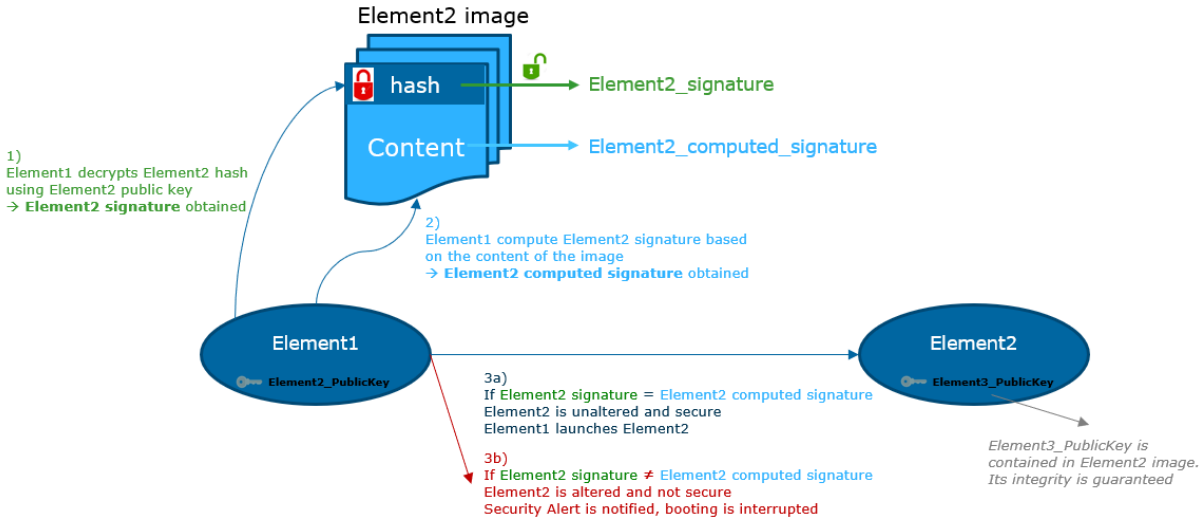


FIGURE 22 - CHAIN OF TRUST PRINCIPLE

The initial element of the chain is critical and called the root of trust.

Note that the only encrypted part of the firmware image is the hash. Encryption of the entire image can be done, but it is answering to a different security threat than "Firmware Integrity".

- Secure Update
  As secure boot allows to guarantee the booted firmwares are conform to the delivered one, this is important to maintain this integrity check while updating. Like for the chain of trust, Secure Update relies on signed image. The signature is computed for the updated image (in RAM), it is compared to the decrypted hash. If there is match, the update image is conform and can be uploaded in the boot memory.

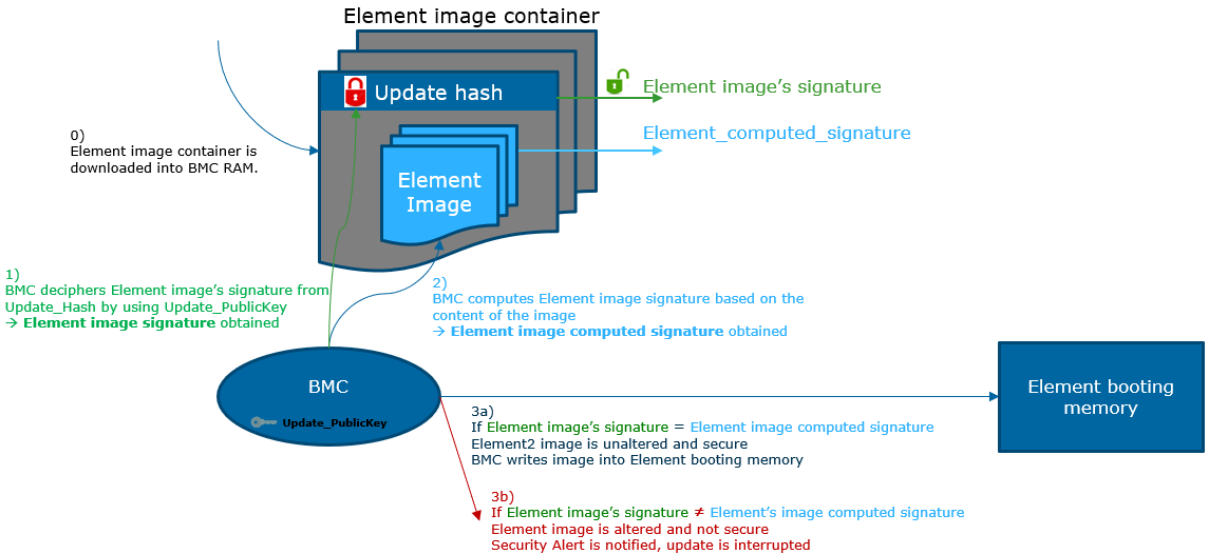


FIGURE 23 - SECURE UPDATE

- HTTPS
- Password Policy
  It seems appropriate than the default administrator password should be different for every customer.

- The blade should implement secure boot based on a hardware Root of Trust on all boot chains (BMC and host)

- The blade should implement secure firmware update for all firmware updates

- BMC SSH Access should be by default de-activated.

- BMC Redfish interface should provide user/password basic login.

- BMC Redfish interface should provide interface to LDAP.

- BMC Redfish interface should provide authentication through TLS certificate.

- BMC Redfish interface should implement Administrator/Operator/Read-only user roles for hierarchical user rights

10 Environmental

For shipping, the OpenSequana blade is shipped containing liquid.

- The blade must resist to shipping ambient air from -10°C to 70°C, with change rate of 25°C/hour.

- The blade must resist to relative humidity from 5% to 95% (not condensing) with a change rate of 30%/hour.

- The blade must be compliant with ETSI EN 300 019-1-2 severity class 2.2 for transportation

Once installed in a BullSequanaXH3000 rack, the blade will experience the following environment:

- The blade must support operating ambient air from +10°C to 55°C with a change rate of 20°C/hour.

- The blade must support relative humidity from 8% to 80% with a change rate of 5%/hour.

- The blade must support water temperature from 21°C to 44°C (see §5 - Thermal interfaces)

In addition, the blade must be certified in accordance to the country where it is expected to be shipped. For example, CE marking is mandatory for selling to a European country (this includes EMI tests, REACH/RHOS3 analysis and other tests).

11 References

11.1External references

The following documents could be provided on demand to partners to detail this design guide.

OpenSequana_SBI - Sideband interface Specification**[**ATOS, **]**

OpenSequana_Redfish - Redfish administration interface Specification**[**ATOS, **]**

OpenSequana_Drawings - Mechanical drawings of an OpenSequana blade**[**ATOS, **]**

OpenSequana_3Dcad - Step file of an OpenSequana blade**[**ATOS, **]**

OpenSequanaPowerModel - LTSpice model of the rack**[**ATOS, **]**