CAHN

Cahn/D4-01-Base-Hardware-Selection

Table Of Contents:

40-D4-01 Base Hardware Selection

Hardware Component Selection

The hardware components need to be considered for the following elements

- 5G Network
- WIFI
- LoraWAN
- Satellite
- NQM Servers

These represent the physical hardware dependencies of the project

5G Network

- 4 uts. Dell PowerEdge R740
- Containing: (AWTS /Amarisoft Integration)
    - 5G full Stack Solution
    - 3 uts. CPRI cards (6 GBIC bays each)
    - 800 GB HD
- 1 ut SD-WAN Core Switch EdgeCore (Switch)
    - DCS204 – Edgecore Networks (edge-core.com)
- 1 ut. Amarisoft CallBOX (PicoCell)
    - https://www.amarisoft.com/test-and-measurement/device-testing/device-products/amari-callbox-classic
- 1 ut. Amarisoft User Equipment Simulator
    - https://www.amarisoft.com/test-and-measurement/network-testing/network-products/amari-ue-simbox-e-series

Radio Heads

- 10 Uts. of AW2S Black Hawk n77 band 43 dBm 5G Radioheads

    - DES000025-PRJ000019-AW2S-BlackHawk_FDD_B3_43dBm_-48V_Specification.pdf

- 10 uts. of AW2S Black Hawk B3 band 4G Radioheads
    - DES000113-PRJ000041-AW2S-Panther_5G_MP_TDD_33dBm_-48V_Specification.pdf

Public WI FI

- 4 uts. cnMatrix TX2028RF-P
    - cnMatrix TX2028RF-P Enterprise Switch | Cambium Networks
- 4 uts. Unifi EdgeSwitch12
    - EdgeSwitch Fiber ES-12F Datasheet (ubnt.com)
- 26 units of Cambium e700 Enterprise Access Points
    - cnPilot e700 Enterprise Outdoor Wi-Fi | Cambium Networks
- 37 uts. Cambium XV2-T0 and XE3 Outdoor Access Points.
    - Cambium XV2-2T XV2-2T Wi-Fi 6 Outdoor Wireless Access Point - English (cambiumnetworks.com)
    - Cambium XE3 XE3-4TN Wi-Fi 6/6E Outdoor Wireless Access Point - English (cambiumnetworks.com)
- Mikrotik RouterBoard R760
    - MikroTik Routers and Wireless - Products: hEX S

LoraWAN

- 1ut. The Things Network Gateway
    - https://www.thethingsindustries.com/docs/gateways/models/dragino-lps8/
- 1ut. Weather Station Sensor SenseCAP S900 + Data Logger S2100
    - https://www.seeedstudio.com/SenseCAPONE-S900-9in1-Compact-Weather-Sensor-p-4881.html
    - https://solution.seeedstudio.com/product/sensor-hub-4g-data-logger/

Satellite

For the Satellite experiments we will be using StarlinQ

https://www.starlink.com/gb/specifications?spec=4

No additional equipment is needed.

NQM Servers

NquiringMinds servers will be run "in house" on commodity server

We have selected **Dell PowerEdge R350** as the target hardware, due to its ubiquity and solid support

https://i.dell.com/sites/csdocuments/Product_Docs/en/Dell-EMC-PowerEdge-R350-Spec-sheet.pdf

Although this is the target hardware, the software is designed so it can run on almost any hardware or cloud deployment

BLACKHAWK

Band 3 20W LTE -48V version

Reference :
DES000025
Author :
N.BREANT
Revision :
D

1. Document

1.1. History

Date	Version	Author	Comments
06/10/2015	A	N.BREANT	Creation draft
30/03/2015	B	J.TASTET	Mechanical Drawing Update
09/03/2017	C	N.BREANT	Specs adjustments
11/11/2017	D	N.BREANT	Doc update

1.2. Approbation

Date	Version	Reviewer(s)	Comments
09/10/2015	A	D.ARNAUD	
30/03/2015	B	N.BREANT	
09/03/2017	C	J.TASTET	
11/11/2017	D	J.TASTET	

2. Introduction

This document is a presentation of AW2S Blackhawk Band 3 20W product version, how to install it and how to connect cables.

2.1. Audience

This document is applicable to the teams charged with designing, developing, integrating, verifying, validating, operating and maintaining the product.
It uses during all the development phases and during CUSTOMER's operations.

2.2. Convention

Symbol	Description
 DANGER	Indicates a hazard with high level of risk, which if is not avoided, will result in death or serious injury
 WARNING	Indicates a hazard with medium or low level of risk, which if not avoided, could result in minor or moderate injury.
 CAUTION	Indicates a potentially hazardous situation, which if is not avoided, could result in equipment damage, data loss, performance degradation, or unexpected results.
	Indicates for commanding a specifically required action.
	Indicates additional information as a reference.

3. Documentary Reference

This chapter provides a list of all the applicable and reference documents, their links with the industrial technical specification as well as the terminology used in drawing them up. Their name and reference designate the documents.

3.1. Applicable documents

Documents listed below are applicable to this DES.

Id.	Title of the document	Reference of the document	Notes / Version
DA1			
DA2			

3.2. Applicable norms and standards

Documents listed below are applicable to this DES.

Id.	Title of the document	Reference of the document	Notes / Version
DS1	LTE Evolved Universal Terrestrial Radio Access (E-UTRA) Base Station (BS) radio transmission and reception	ETSI TS 136104	V12.5.0
DS2	Classification of Environnemental Conditions Storage	ETSI EN 300 019-1-1	V2.1.4 (2003-04)
DS3	Classification of Environnemental Conditions Transportation.	ETSI EN 300 019-1-2	V2.2.1 (2014-04)
DS4	Classification of Stationary use at non-weather protected locations	ETSI EN 300 019-1-4	V2.2.1 (2014-04)
DS5	Specification of environmental tests; Storage	ETSI EN 300 019-2-1	V2.1.6 (2014-06)
DS6	Specification of environmental tests; Conditions Transportation.	ETSI EN 300 019-2-2	V2.2.1 (2011-11)
DS7	Specification of environmental tests; use at non-weather protected locations	ETSI EN 300 019-2-4	V2.3.1 (2012-12)
DS8	Degrees of Protection Provided by Enclosures – IP Code.	IEC 60529	
DS9	Safety of information technology equipment.	EN 60950	Ed.2001
DS10	Standard for Safety of information Technology Equipment-including Electrical Business Equipment	UL 60950	Ed.3 2000

Id.	Title of the document	Reference of the document	Notes / Version
DS11	Test for flammability of materials for parts in devices and appliances.	UL94	
DS12	Safety of Information Technology Equipment	CSA C22.2 No 60950	Ed.3 2000
DS13	Common Public Radio Interface (CPRI); Interface Specification	V6.0	2013-08
DS14	Open Radio equipment Interface (ORI); Requirements for Open Radio equipment Interface (ORI) (Release 1)	ETSI GS ORI 001	V4.1.1
DS15	Open Radio equipment Interface (ORI); ORI Interface Specification; Part 1: Low Layer	ETSI GS ORI 002-1	V4.1.1
DS16	Open Radio equipment Interface (ORI); ORI Interface Specification; Part 2: Control and Management	ETSI GS ORI 002-2	V4.1.1
DS17	ElectroMagnetic Compatibility (EMC)standard for radio equipment and services; Part 50: Specific conditions for Cellular Communication Base Station (BS), repeater and ancillary equipment; Harmonised Standard covering the essential requirements of article 3.1(b) of Directive 2014/53/EU	ETSI EN 301 489-50	2.1.1
DS18	ROHS2 requirements	2011/65/EU	27 January 2003
DS19	WEEE requirements	2012/19/EU	27 January 2003
DS20	REACH requirements	n°1907/2006/CE	SVCH 06/2014

3.3. Reference documents

These are documents which have no contractual nature and which have been the basis for drawing up the document.

Id.	Title of the document	Reference of the document	Notes / Version
DR1			
DR2			

4. Terminology

4.1. Abbreviations

CFR	Crest Factor Reduction
CPRI	Common Public Radio Interface
DPD	Digital Pre-Distortion
EVM	Error Vector Magnitude
IMD	Intermodulation
MCPA	Multi-Carrier Power Amplifier
PA	Power Amplifier
PAR	Power to Average Ratio
RF	Radio Frequency
SCPA	Single Carrier Power Amplifier
SEM	Spectral Emission Mask

4.2. Definitions

Crest Factor Reduction

This is a technic to reduce the power to average ratio while keeping EVM and SEM in specification.

Digital Pre-Distortion

This is a technic to enhance power amplifier linearity.

Power to Average Ratio

This is the Peak power over average power ratio.

5. Safety concern

The purpose of this section is to ensure the safety of users and prevent property damage. Please read this document carefully for proper use.

5.1. Power and grounding

DANGER	***Watches, Rings and other Metallic accessories*** Do not wear accessories such as watches and rings in order to prevent electrical shock.

	Power switch off Make sure the power switch of power supply is off when installing the system. Installing with power switch on may cause system damage of fatal human injury when cables are not correctly connected.

WARNING	***Warning for connecting the ground cable*** In cabling, the connection of cables without the connection of the ground cable may cause the damage of the equipment or the injury of the worker. Connect the ground cable first.

5.2. Installation

DANGER	***Warning for Laser Beam Running through Optical Cables*** In the system, the laser beam emitting light runs through the optical cable. The exposure of the laser beam on worker's eye may cause serious injury so that it should be handled with care. 

DANGER	***Protection gloves and goggles*** Make sure that worker wears protection gloves and goggles to prevent damages from debris while drilling holes in a wall or ceiling 

5.3. Power and Feeder Line

CAUTION	***Cautions while cleaning power supply*** While cleaning the power supply device, take caution that the device does not come in contact with alien bodies that may cause power failure.

CAUTION	***Handling the Power Cable*** - Handling the power cable incorrectly may damage the rack or cause an electric shock through the cable. Ensure the power switch on the rectifier or the system is turned off before handling the power calbe. - The fixing materials for power cable must be tightly secured to prevent electrical accidents.

CAUTION	***Precautions for measuring insulation resistance*** Since high voltage is used for measuring insulation resistance, insulation resistance should not be measured when the system is in operation. Make sure to only measure the insulation resistance of the appointed part. Other components such as the system's internal components and the unit (system frame), components of the communication cables, units, etc. should not be measured.

	**Cable work sequence** When performing cable work for the system, proceed with the ground work before any other work to prevent errors occurring due to static electricity and other reasons.

 CAUTION	**Connection of Feeder Cable Connector** Connecting the feeder cable connector is critical process, so the qualified workers who finished the related education should perform.

 CAUTION	**Radius of curvature of feeder line** When installing a feeder line, the radius of curvature of the sections where cables bent should be larger than the allowed radius of curvature. If the radius of curvature for the feeder line installation is less than the allowed radius curvature, it may affect the performance of the system.

 CAUTION	**Feeder cable and Antenna PIM** The installed antenna and feeder cable need to have an intermodulation level lower than -153dBc. Feeder return loss should be better than 20dB in the band. Antenna return loss should be better than 15dB. If the intermodulation or return loss is higher than allowed values, it may affect the performance of the system.

5.4. Installation and de-installation

 CAUTION	**Caution while cleaning RRH** Make sure that worker does not damage installed cable while cleaning the RRH

 CAUTION	**System installation and access** Only authorized and trained workers are allowed to install or access the system.

 DANGER	**Do not work by yourself** Worker must not work alone in any key process.

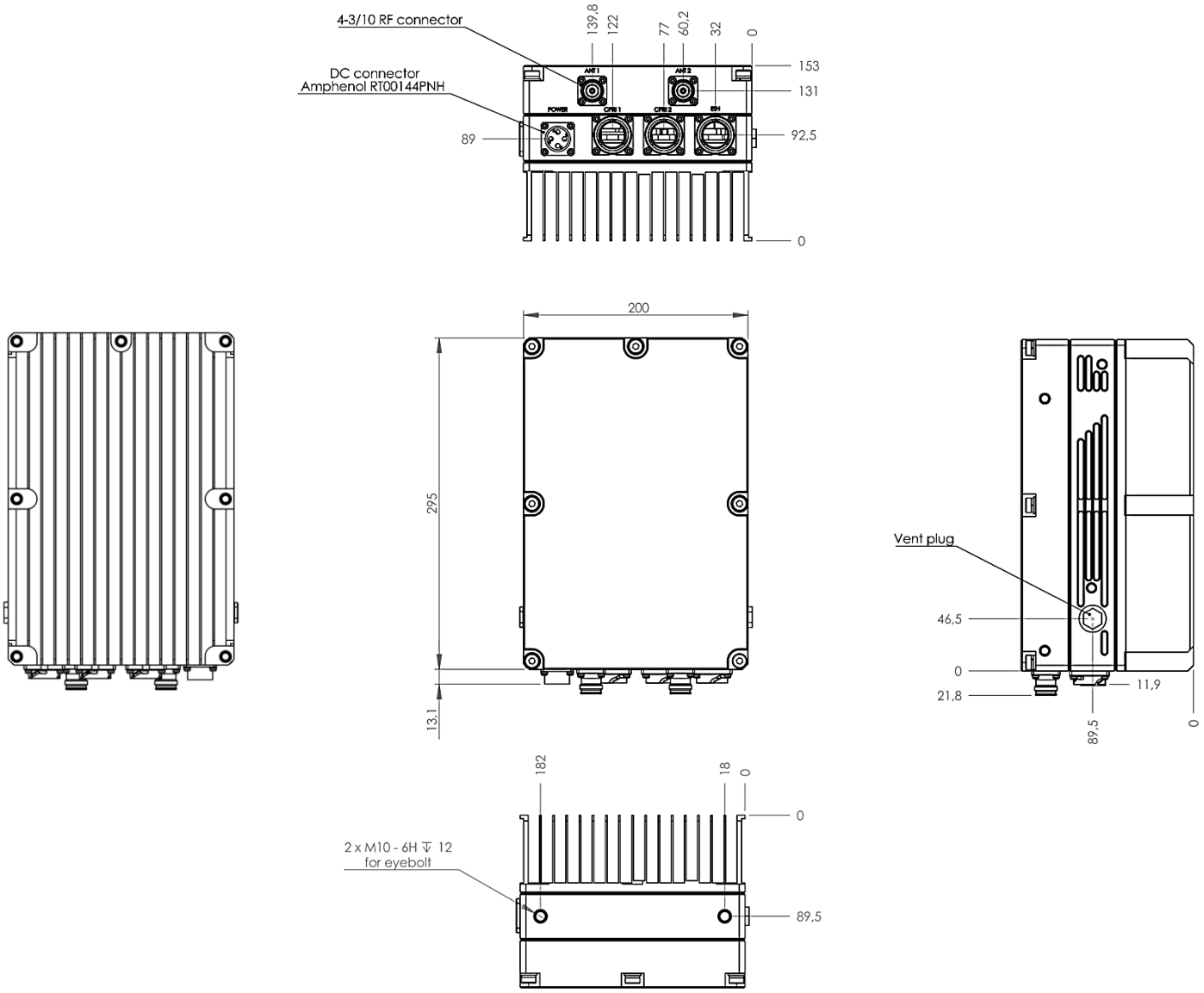
 CAUTION	**Management of unused ports** Cover the unused ports (conduit, cable, gland, etc.) with waterproff cap (sealing cap) to prevent infiltration of foreign material such as dust, moisture, bug or water.

 CAUTION	**Caution when connecting the optical cable** When connecting the optical cable, be careful to keep the cutting section of the connector core away from dust and foreign substances. If the cable is soiled with foreign substances, do not blow on the cable to remove them. Make sure to remove dust or foreign substances in accordance with the cleaning instructions provided by the connector manufacturer.

 CAUTION	**Installing the antenna** When you install the antenna, the distance and angle between the antenna and the lightning rod must be within the protective angle (left/right side 45° each from the central axis) to prevent the antenna for lightning damage.

6. Before Installation

6.1. System configuration and structure



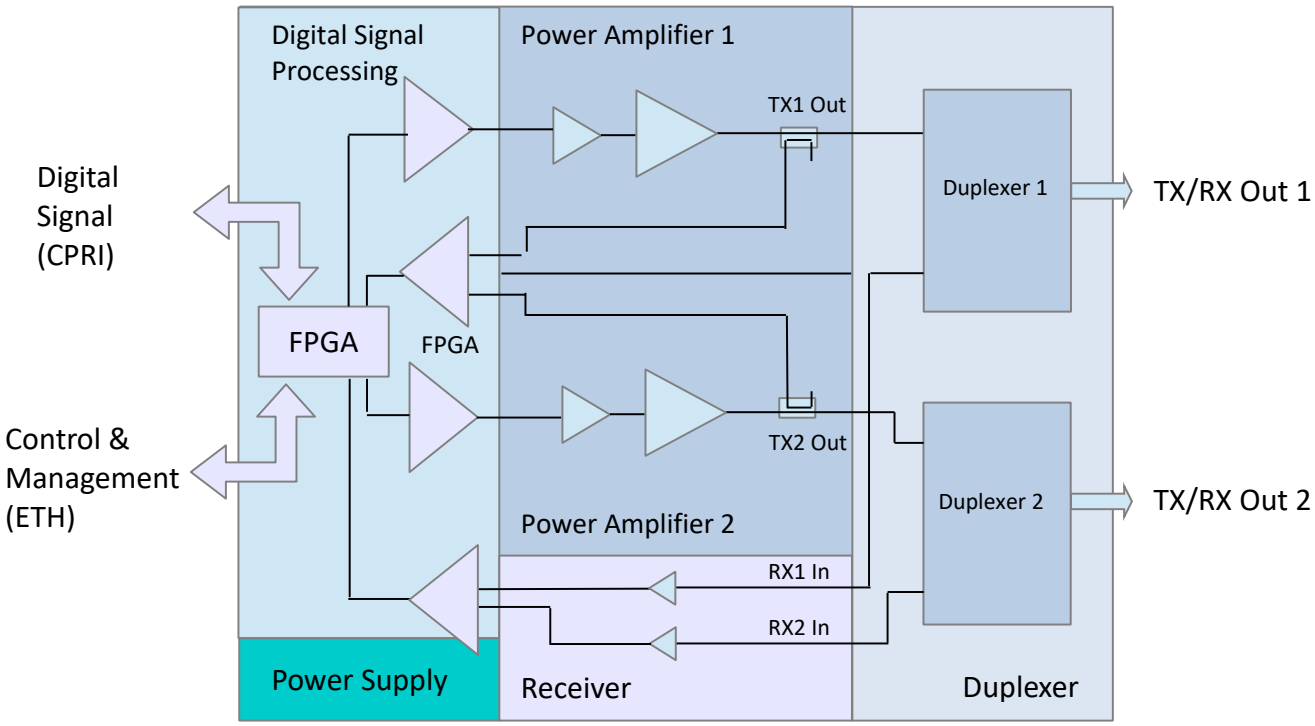
Dimensions are in mm

7. Product Description

7.1. Platform Specifications

Parameters	Value	Units
Tx/Rx Ports	2	
Max. Nb Carriers per TX/RX	2 (1.4, 3, 5, 10, 15, 20MHz)	
Tx Frequency range	1805 to 1880	MHz
Rx Frequency range	1710 to 1785	MHz
Tx Max Pout	20	W Avg.
Power Supply	Isolated DC -36 to -58	V
Power Consumption	220	W typ.
Weight	<12	Kg typ.
Dimensions	295x200x138	mm
I/Q connectivity	2 CPRI V6.0	
Local Management & debugging	1 Gigabit Eth.	

7.2. AW2S Platform bloc Diagram



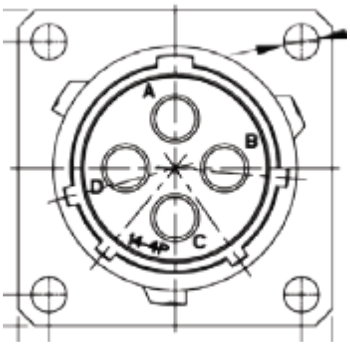
This is the basic diagram of the product.

8. General

Requirement	Parameter
GEN-000	Platform can be easily shifted in frequency - Use of Power pallet dimensioned for lowest frequency - Wide band transceiver - Low power stage using large frequency range
GEN-001	Support 2 Tx and 2 Rx in the products to avoid using two products to do LTE MIMO 2x2
GEN-002	Try to support up 100MHz of DPD bandwidth to allow 2 LTE 20MHz carriers
GEN-003	Keep dimension as low as possible
GEN-004	Outdoor product IP67 when installed, limit joint to 2
GEN-005	Status needs to give a clear understanding of the state of the unit, in particular if the unit is transmitting.

10. Connectivity

Requirement	Parameter
CON-000	2 CPRI Links up to line rate 6 Ref. 20
CON-001	LTE user plane following E-UTRA mapping ORI Ref.21 or AW2S broadband Mapping
CON-002	LTE user plane can as well support MIMO interleaving
CON-003	Local C&M and Debug using Serial port and Gigabit Ethernet
CON-004	Power supply connector Amphenol reference RT0014-4PNH



A	GROUND
B	0V
C	UNCONNECTED
D	-48V

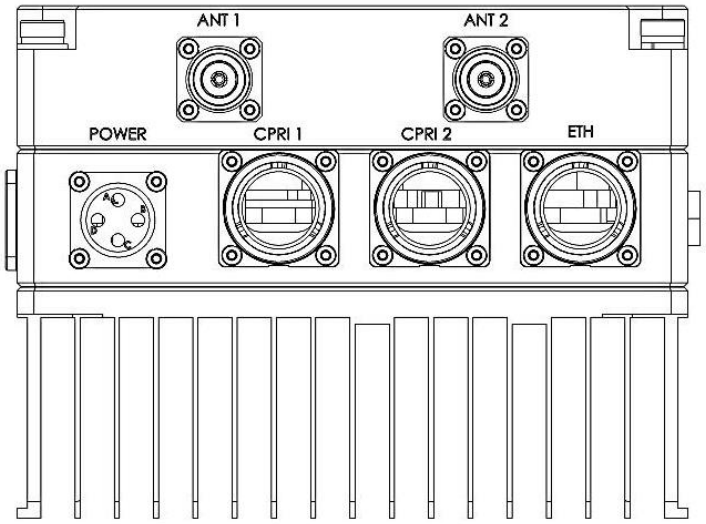
Amphenol reference RT06144SNH can be use as plug depending of the gauge used, however AWG-12 minimum is recommended for long cables, ref SS12A1T. Cable grip reference used is the RT0L-14CG-S1 or –S2.

CON-005	Fiber cable connection is done through SFP+ cages with R2CT connector for ingress protection

The R2CT plug reference R2CT 115 000 can be use from Amphenol or Radiall.

CON-006	Gigabit Ethernet connection is RJ45 connector with R2CT connector for ingress protection

The R2CT plug reference R2CT 127 000 can be use from Amphenol or Radiall.



ANT 1	Antenna port 1	4.3/10 Female
ANT 2	Antenna port 2	4.3/10 Female
Power	Power supply	RT00144PNH
CPRI 1	CPRI port 1	SFP+ cage with R2CT socket
CPRI 2	CPRI port 2	SFP+ cage with R2CT socket
ETH	GBEthernet Debug port	RJ45 with R2CT socket

11. Power Supply

Requirement	Parameter	Value	Unit	Comments
PSU-000	Operating Voltage Min.	-36	V	
PSU-001	Operating Voltage Max.	-58	V	
PSU-002	Max consumption	225	W	At Pout 20W RMS for each Tx
PSU-003	Max Isolation	1500	V	

12. Mechanical

Requirement	Parameter	Value	Unit	Comments
MEC-000	The product shall fit in a IP67 outdoor passive cooling package			
MEC-001	Height	153	mm	

MEC-002	Depth	200	mm	
MEC-003	Length	295	mm	
MEC-004	Weight	<12	kg	

13. Transmitter

Requirement	Parameter	Value	Unit	Comments
TX-LTE-000	Number of Carriers	1		
TX-LTE-001	Max. Output power	43	dBm	
TX-LTE-002	Min. Output power	Max – 25dB	dBm	
TX-LTE-003	Power Precision	+/-0.5	dBm	10-40°C
TX-LTE-004	Power Precision	+/-0.75	dBm	Other Temperature
TX-LTE-005	Power Step	1+/-0.2	dB	
TX-LTE-006	Channel Bandwidth	1.4/3/5/ 10/15/20	MHz	
TX-LTE-007	Channel Offset	200	kHz	
TX-LTE-008	TxOFF residual noise	<-145	dBm/Hz	
TX-LTE-009	EVM 64QAM	<8	%	
TX-LTE-010	EVM 16QAM	<12.5	%	
TX-LTE-011	EVM QPSK	<17.5	%	
TX-LTE-012	Spurious (9KHz-150KHz)	<-36	dBm	1KHz Bandwidth
TX-LTE-013	Spurious (150KHz-30MHz)	<-36	dBm	10KHz Bandwidth
TX-LTE-014	Spurious (30MHz-1GHZ)	<-36	dBm	100KHz Bandwidth
TX-LTE-015	Spurious (1GHz-12.5GHZ)	<-30	dBm	1MHz Bandwidth
TX-LTE-016	Spectrum Emission mask			Category B (Option 2)
TX-LTE-017	ACLR	>50	dBc	No need to be better than -18dBm/1MHz

14. Receiver

Requirement	Parameter	Value	Unit	Comments
RX-LTE-000	Number of Carriers	1		
RX-LTE-001	Channel Bandwidth	1.4/3/5/ 10/15/20	MHz	
RX-LTE-002	Channel Offset	200	kHz	
RX-LTE-003	Max. Input power	>-35dBm	dBm	Composite Power
RX-LTE-004	Reference Sensitivity 5MHz	<-101.5	dBm	
RX-LTE-005	Reference Sensitivity 10MHz	<-101.5	dBm	
RX-LTE-006	Reference Sensitivity 20MHz	<-101.5	dBm	
RX-LTE-007	Dynamic Sensitivity 5MHz	<-70.2	dBm	Interferer -82.5dBm
RX-LTE-008	Dynamic Sensitivity 10MHz	<-70.2	dBm	Interferer -79.5dBm
RX-LTE-009	Dynamic Sensitivity 20MHz	<-70.2	dBm	Interferer -79.5dBm
RX-LTE-010	E-UTRA Narrow band Adj. Channel Selectivity 5MHz[1]	Pref + 6dB	dBm	Interferer -49dBm
RX-LTE-011	E-UTRA Narrow band Adj. Channel Selectivity 10MHz[1]	Pref + 6dB	dBm	Interferer -49dBm
RX-LTE-012	E-UTRA Narrow band Adj. Channel Selectivity 20MHz[1]	Pref + 6dB	dBm	Interferer -49dBm
RX_LTE-013	E-UTRA Adj. Channel Selectivity 5MHz[1]	Pref + 6dB	dBm	Interferer -52dBm
RX_LTE-014	E-UTRA Adj. Channel Selectivity 10MHz[1]	Pref + 6dB	dBm	Interferer -52dBm
RX_LTE-015	E-UTRA Adj. Channel Selectivity 20MHz[1]	Pref + 6dB	dBm	Interferer -52dBm

Note 1. See Ref 1§7.5 for interferer characteristics

Requirement	Parameter	Value	Unit	Comments
RX-LTE-018	Generic Blocking[1]	-43	dBm	+20/-20MHz
RX-LTE-019	Generic Blocking[1]	-15	dBm	+CW further
RX-LTE-020	Intermodulation[2]	-52	dBm	Pref + 6dB

Note 1. See Ref 1§7.6.1 for interferer characteristics
Note 2. See Ref 1§7.8.1 for measurement characteristics

15. Control & Management

Requirement	Parameter
CMM-000	Control & Management based on ORI specification Ref. 22
CMM-001	Supported Object: - RE - TxSigPath_EUTRA - RxSigPath_EUTRA - oriLink - antPort
CMM-002	Supported Device Management Request: - HealthCheck - set Time - RE Reset
CMM-003	Supported Software Management Request : - Version Query - Software Update Preparation - Software Download - Software Activation
CMM-004	Support Fault Management

16. Environmental

Requirement	Parameter	Value	Unit	Comments
ENV-000	Operating Temp. Range Min.	-40	°C	Ref. 4 Class4.1
ENV-001	Operating Temp. Range Max.	+55 (*)	°C	Ref. 4 Class4.1 (with sun cover)
ENV-002	Storage Temp. Range Min.	-40	°C	Ref. 2 Class1.2
ENV-003	Storage Temp. Range Max.	+70	°C	Ref. 2 Class1.2
ENV-004	Transportation Temp. Range Min.	-40	°C	Ref. 3 Class2.3
ENV-005	Transportation Temp. Range Max.	+70	°C	Ref. 3 Class2.3
ENV-006	Shall respect at least IP67 as defined by the document Ref. 8 when in approved operational condition			
ENV-007	Shall not resonate in audibled range (20Hz-20KHz) when in approved operational condition			

*: Operational max. temperature can be extended to 60°C in specific case (output power derating will be applied for ex.). Contact us for further information.

17. Regulatory

Requirement	Parameter
REG-000	Product shall pass and have the CE marking
REG-001	For electrical safety product shall comply to requirement defined per Ref. 10
REG-002	Product shall comply to requirement defined per Ref. 11 & 12
REG-003	ROHS : the product and its internal components shall have to fulfill the requirements Ref. 18
REG-004	WEEE : the product and its internal components shall have to fulfill the requirements of Ref. 19
REG-005	REACH: AW2S shall fulfill at any time all requirements according to the regulation Ref 20 concerning the handling of chemical substances. AW2S shall especially fulfill all duties imposed upon him according to Articles 31 to 33 (incl.) and shall provide all information which the customer may require.

PANTHER 4X4 MIMO RRH

MEDIUM POWER 5G NR & LTE

LTE/NR -48V version

Reference :
DES000113

Author :
D.ARNAUD

Revision :
A

1. Document

1.1. History

Date	Version	Author	Comments
11/08/2020	A	D.ARNAUD	Creation

1.2. Approbation

Date	Version	Reviewer(s)	Comments

2. Introduction

This document is a presentation of AW2S Panther Band NR n78/n77 and LTE B42 & B43 2W product version, how to install it and how to connect cables.

2.1. Audience

This document is applicable to the teams charged with designing, developing, integrating, verifying, validating, operating and maintaining the product.
It uses during all the development phases and during CUSTOMER's operations.

2.2. Convention

Symbol	Description
 DANGER	Indicates a hazard with high level of risk, which if is not avoided, will result in death or serious injury
 WARNING	Indicates a hazard with medium or low level of risk, which if not avoided, could result in minor or moderate injury.
 CAUTION	Indicates a potentially hazardous situation, which if is not avoided, could result in equipment damage, data loss, performance degradation, or unexpected results.
	Indicates for commanding a specifically required action.
	Indicates additional information as a reference.

3. Documentary Reference

This chapter provides a list of all the applicable and reference documents, their links with the industrial technical specification as well as the terminology used in drawing them up. Their name and reference designate the documents.

3.1. Applicable documents

Documents listed below are applicable to this DES.

Id.	Title of the document	Reference of the document	Notes / Version
DA1			
DA2			

3.2. Applicable norms and standards

Documents listed below are applicable to this DES.

Id.	Title of the document	Reference of the document	Notes / Version
DS1	LTE Evolved Universal Terrestrial Radio Access (E-UTRA) Base Station (BS) radio transmission and reception	ETSI TS 136104	V15.8.0
DS2	5G;NR; Base Station (BS) radio transmission and reception	ETSI TS 138104	V15.8.0
DS3	Classification of Environnemental Conditions Storage	ETSI EN 300 019-1-1	V2.1.4 (2003-04)
DS4	Classification of Environnemental Conditions Transportation.	ETSI EN 300 019-1-2	V2.2.1 (2014-04)
DS5	Classification of Stationary use at non-weather protected locations	ETSI EN 300 019-1-4	V2.2.1 (2014-04)
DS6	Specification of environmental tests; Storage	ETSI EN 300 019-2-1	V2.1.6 (2014-06)
DS7	Specification of environmental tests; Conditions Transportation.	ETSI EN 300 019-2-2	V2.2.1 (2011-11)
DS8	Specification of environmental tests; use at non-weather protected locations	ETSI EN 300 019-2-4	V2.3.1 (2012-12)
DS9	Degrees of Protection Provided by Enclosures – IP Code.	IEC 60529	
DS10	Safety of information technology equipment.	EN 60950	Ed.2001
DS11	Standard for Safety of information Technology Equipment-including Electrical Business Equipment	UL 60950	Ed.3 2000

Id.	Title of the document	Reference of the document	Notes / Version
DS12	Test for flammability of materials for parts in devices and appliances.	UL94	
DS13	Safety of Information Technology Equipment	CSA C22.2 No 60950	Ed.3 2000
DS14	Common Public Radio Interface (CPRI); Interface Specification	V6.0	2013-08
DS15	Open Radio equipment Interface (ORI); Requirements for Open Radio equipment Interface (ORI) (Release 1)	ETSI GS ORI 001	V4.1.1
DS16	Open Radio equipment Interface (ORI); ORI Interface Specification; Part 1: Low Layer	ETSI GS ORI 002-1	V4.1.1
DS17	Open Radio equipment Interface (ORI); ORI Interface Specification; Part 2: Control and Management	ETSI GS ORI 002-2	V4.1.1
DS18	Electromagnetic compatibility and Radio spectrum Matters (ERM); ElectroMagnetic Compatibility (EMC) standard for radio equipment and services	ETSI EN 301 489-23	1.5.1
DS19	ROHS2 requirements	2011/65/EU	27 January 2003
DS20	WEEE requirements	2012/19/EU	27 January 2003
DS21	REACH requirements	n°1907/2006/CE	SVCH 06/2014

3.3. Reference documents

These are documents which have no contractual nature and which have been the basis for drawing up the document.

Id.	Title of the document	Reference of the document	Notes / Version
DR1			
DR2			

4. Terminology

4.1. Abbreviations

CFR	Crest Factor Reduction
CPRI	Common Public Radio Interface
DPD	Digital Pre-Distortion
EVM	Error Vector Magnitude
IMD	Intermodulation
MCPA	Multi-Carrier Power Amplifier
PA	Power Amplifier
PAR	Power to Average Ratio
RF	Radio Frequency
SCPA	Single Carrier Power Amplifier
SEM	Spectral Emission Mask

4.2. Definitions

Crest Factor Reduction

This is a technic to reduce the power to average ratio while keeping EVM and SEM in specification.

Digital Pre-Distortion

This is a technic to enhance power amplifier linearity.

Power to Average Ratio

This is the Peak power over average power ratio.

5. Safety concern

The purpose of this section is to ensure the safety of users and prevent property damage. Please read this document carefully for proper use.

5.1. Power and grounding

 DANGER	***Watches, Rings and other Metallic accessories*** Do not wear accessories such as watches and rings in order to prevent electrical shock.

	**Power switch off** Make sure the power switch of power supply is off when installing the system. Installing with power switch on may cause system damage of fatal human injury when cables are not correctly connected.

 WARNING	***Warning for connecting the ground cable*** In cabling, the connection of cables without the connection of the ground cable may cause the damage of the equipment or the injury of the worker. Connect the ground cable first.

5.2. Installation

 DANGER	***Warning for Laser Beam Running through Optical Cables*** In the system, the laser beam emitting light runs through the optical cable. The exposure of the laser beam on worker's eye may cause serious injury so that it should be handled with care. 

DANGER	***Protection gloves and goggles*** Make sure that worker wears protection gloves and goggles to prevent damages from debris while drilling holes in a wall or ceiling 

5.3. Power and Feeder Line

CAUTION	***Cautions while cleaning power supply*** While cleaning the power supply device, take caution that the device does not come in contact with alien bodies that may cause power failure.

CAUTION	***Handling the Power Cable*** - Handling the power cable incorrectly may damage the rack or cause an electric shock through the cable. Ensure the power switch on the rectifier or the system is turned off before handling the power calbe. - The fixing materials for power cable must be tightly secured to prevent electrical accidents.

CAUTION	***Precautions for measuring insulation resistance*** Since high voltage is used for measuring insulation resistance, insulation resistance should not be measured when the system is in operation. Make sure to only measure the insulation resistance of the appointed part. Other components such as the system's internal components and the unit (system frame), components of the communication cables, units, etc. should not be measured.

	**Cable work sequence** When performing cable work for the system, proceed with the ground work before any other work to prevent errors occurring due to static electricity and other reasons.

CAUTION	**Connection of Feeder Cable Connector** Connecting the feeder cable connector is critical process, so the qualified workers who finished the related education should perform.

 CAUTION	**Radius of curvature of feeder line** When installing a feeder line, the radius of curvature of the sections where cables bent should be larger than the allowed radius of curvature. If the radius of curvature for the feeder line installation is less than the allowed radius curvature, it may affect the performance of the system.

 CAUTION	**Feeder cable and Antenna PIM** The installed antenna and feeder cable need to have an intermodulation level lower than -153dBc. Feeder return loss should be better than 20dB in the band. Antenna return loss should be better than 15dB. If the intermodulation or return loss is higher than allowed values, it may affect the performance of the system.

5.4. Installation and de-installation

 CAUTION	**Caution while cleaning RRH** Make sure that worker does not damage installed cable while cleaning the RRH

 CAUTION	**System installation and access** Only authorized and trained workers are allowed to install or access the system.

 DANGER	**Do not work by yourself** Worker must not work alone in any key process.

 CAUTION	**Management of unused ports** Cover the unused ports (conduit, cable, gland, etc.) with waterproff cap (sealing cap) to prevent infiltration of foreign material such as dust, moisture, bug or water.

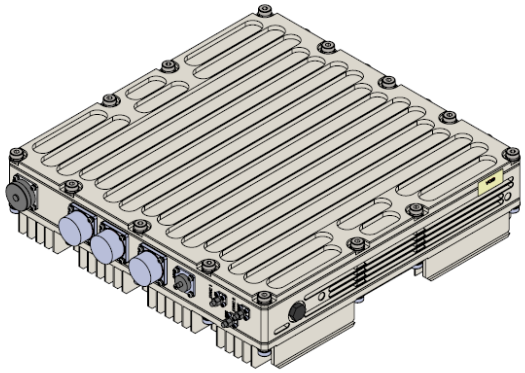
 CAUTION	**Caution when connecting the optical cable** When connecting the optical cable, be careful to keep the cutting section of the connector core away from dust and foreign substances. If the cable

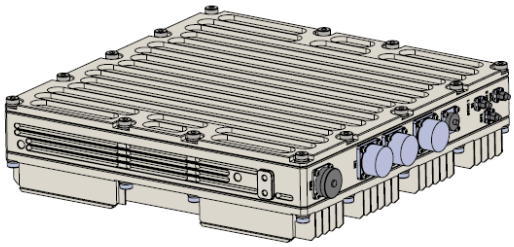
	is soiled with foreign substances, do not blow on the cable to remove them. Make sure to remove dust or foreign substances in accordance with the cleaning instructions provided by the connector manufacturer.

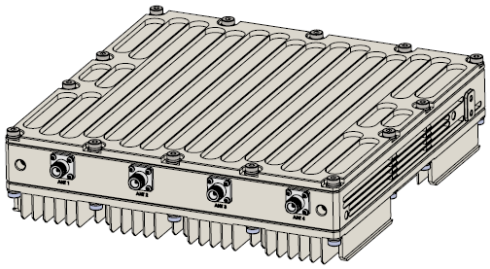
 CAUTION	**Installing the antenna** When you install the antenna, the distance and angle between the antenna and the lightning rod must be within the protective angle (left/right side 45° each from the central axis) to prevent the antenna for lightning damage.

6. Before Installation

6.1. System configuration and structure







7. Product Description

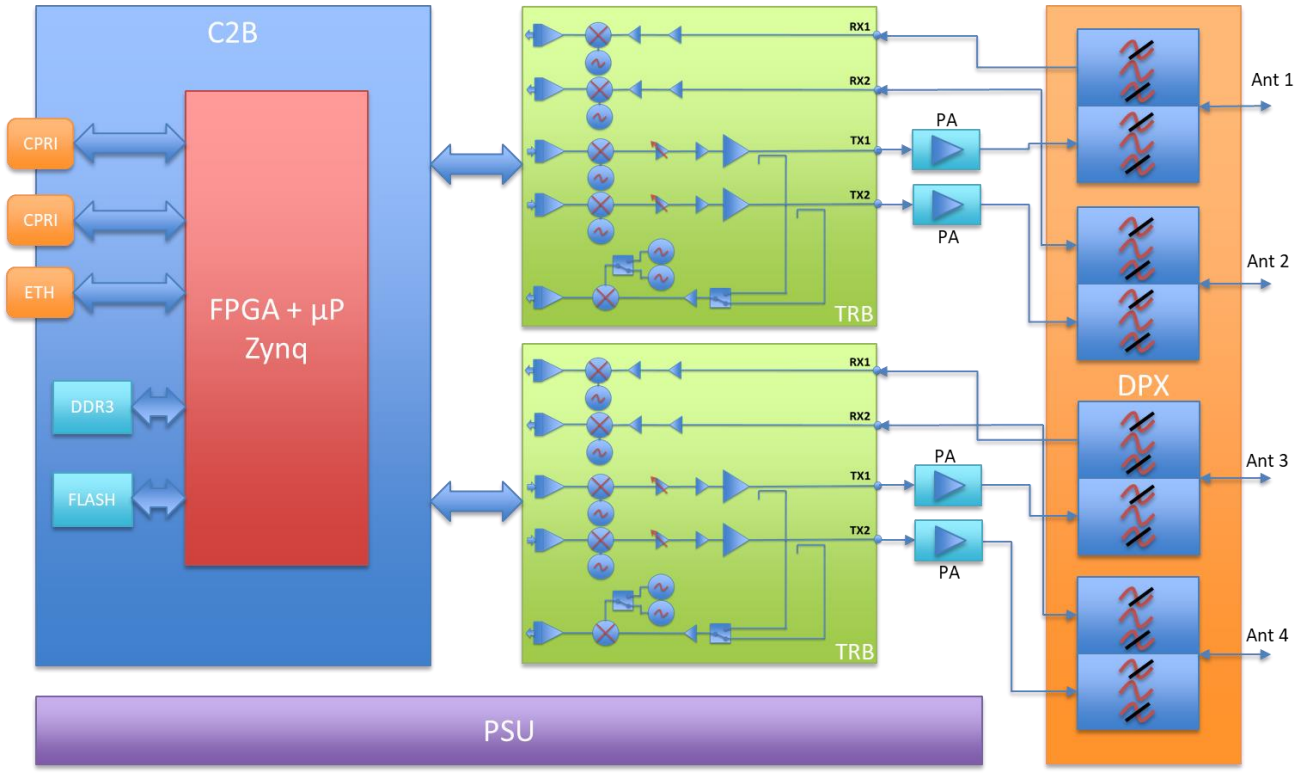
7.1. Model List

Frequency (MHZ)	Nr Band	Lte Band
3400-3800	N78	B42/B43
3800-4100	N77	

7.2. Platform Specifications

Parameters	*Value*	*Units*
Tx/Rx Ports	**4**	
Max. Nb Carriers per TX/RX	**2 LTE (5,10,15,20MHz)** **1 5G-NR (5 to 100MHz)**	**Or 1 LTE and 1 NR**
Tx Max Pout	**2**	**W Avg.**
Power Supply	**Isolated DC -40 to -58**	**V**
Power Consumption	**250**	**W typ.**
Weight	**<15**	**Kg Max.**
Dimensions	**370x369.2x91.3**	**mm**
I/Q connectivity	**2 CPRI up to rate 8**	
Local Management & debugging	**1 Gigabit Eth.**	

7.3. AW2S Platform block Diagram



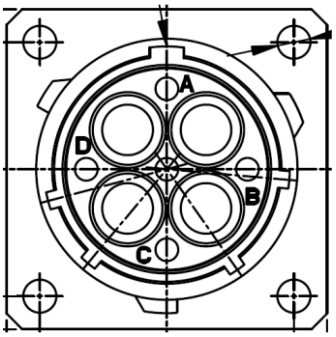
This is the basic diagram of the product.

8. General

Requirement	Parameter
GEN-000	Platform can be easily shifted in frequency
GEN-001	Support 4 Tx and 4 Rx in the products to avoid using two products to do MIMO 4x4
GEN-002	Keep dimension as low as possible
GEN-003	Outdoor product IP66 when installed and all ports connected.
GEN-004	Status needs to give a clear understanding of the state of the unit, in particular if the unit is transmitting.

9. Connectivity

Requirement	Parameter
CON-000	2 CPRI Links up to line rate 8 Ref. 20
CON-001	LTE user plane following E-UTRA mapping ORI Ref.21 or AW2S broadband Mapping
CON-002	LTE user plane can as well support MIMO interleaving
CON-003	Local C&M and Debug using Serial port and Gigabit Ethernet
CON-004	Power supply connector Amphenol reference RT00164PNH



A	GROUND
B	UNCONNECTED
C	0V
D	-48V

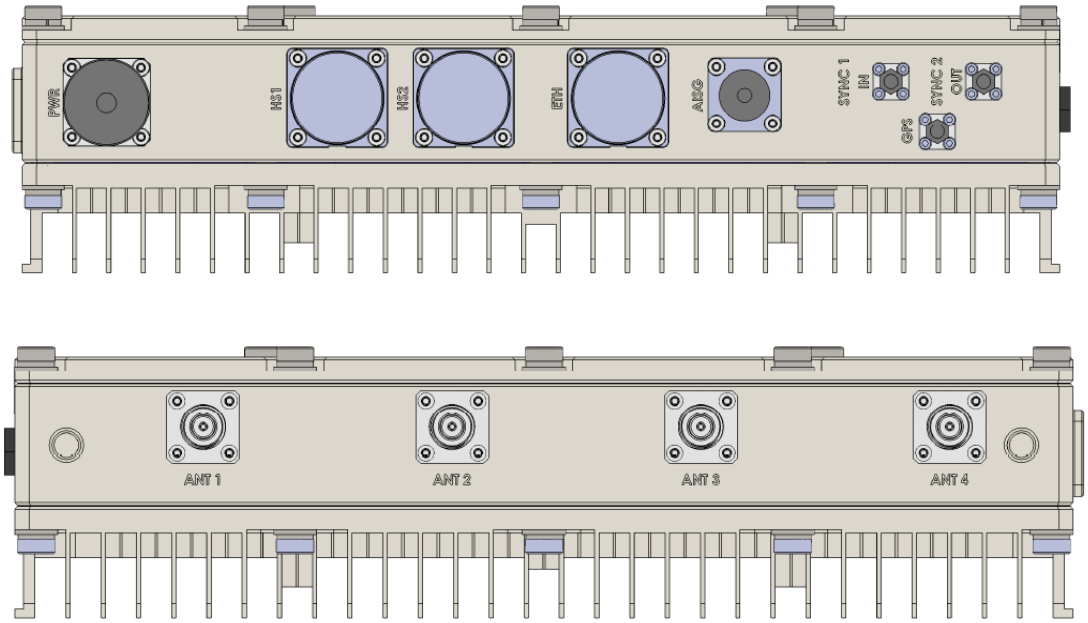
Amphenol reference RT06164SNH can be use as plug depending of the gauge used, however AWG-12 minimum is recommended for long cables, ref MS10A23S. Cable grip reference used is the RT0L-16CG-S1 or –S2.

CON-005	Fiber cable connection is done through SFP+ cages with R2CT connector for ingress protection

The R2CT plug reference R2CT 115 000 can be use from Amphenol or Radiall.

CON-006	Gigabit Ethernet connection is RJ45 connector with R2CT connector for ingress protection

The R2CT plug reference R2CT 127 000 can be use from Amphenol or Radiall.



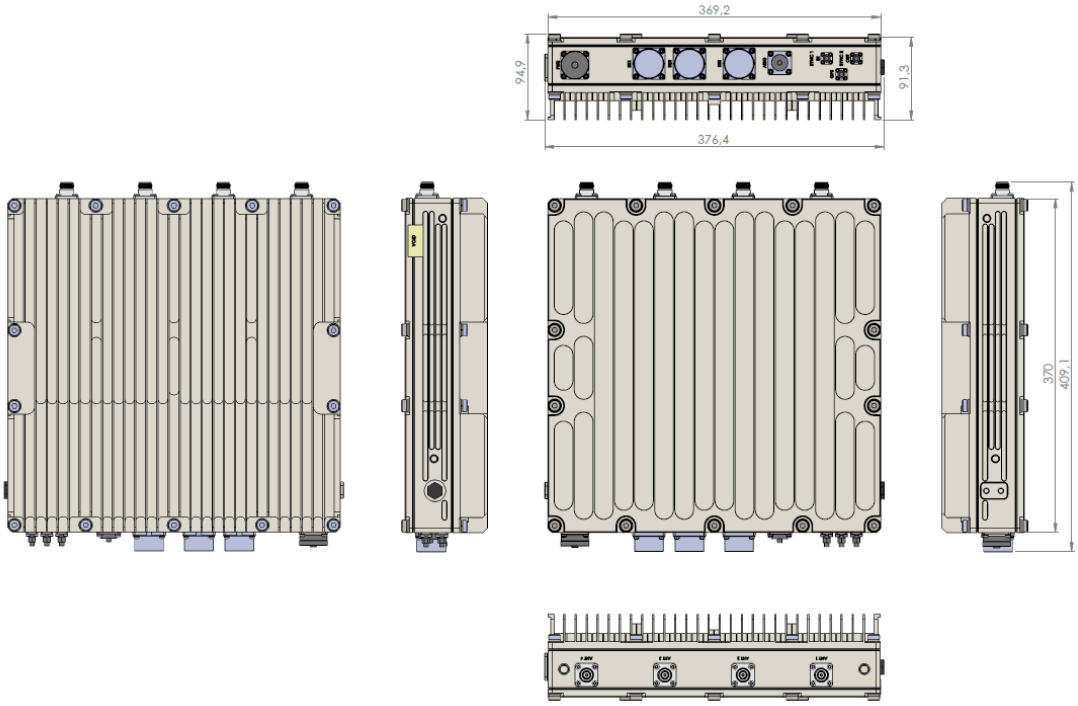
ANT 1	Antenna port 1	N Female
ANT 2	Antenna port 2	N Female
ANT 3	Antenna port 3	N Female
ANT 4	Antenna port 4	N Female
PSU	Power supply	RT06164PNH
CPRI 1	CPRI port 1	SFP+ cage with R2CT socket
CPRI 2	CPRI port 2	SFP+ cage with R2CT socket
ETH	GBEthernet Debug port	RJ45 with R2CT socket
GPS	GPS port	SMA female
Sync1	Sync In port	SMA female
Sync2	Sync out port	SMA female

10. Power Supply

Requirement	Parameter	Value	Unit	Comments
PSU-000	Operating Voltage Min.	-40	V	
PSU-001	Operating Voltage Max.	-58	V	
PSU-002	Typical consumption	130	W	At Pout 2W RMS for each Tx
PSU-003	Max Isolation	1500	V	

11. Mechanical

Requirement	Parameter	Value	Unit	Comments
MEC-000	The product shall fit in a IP66 outdoor passive cooling package			
MEC-001	Height	91.3	mm	
MEC-002	Depth	369.2	mm	
MEC-003	Length	370	mm	
MEC-004	Weight	<15	kg	



Dimensions are in mm

12. Transmitter

Requirement	Parameter	Value	Unit	Comments
TX-GEN-000	Number of Carriers	2x LTE or 1x NR		
TX-GEN-001	Max. Output power	33	dBm	
TX-GEN-002	Min. Output power	Max – 15dB	dBm	
TX-GEN-003	Power Precision	+/-0.5	dBm	10-40°C
TX-GEN-004	Power Precision	+/-0.75	dBm	Other Temperature
TX-GEN-005	Power Step	1+/-0.2	dB	

12.1. LTE

Requirement	Parameter	Value	Unit	Comments
TX-LTE-000	Number of Carriers	2		
TX-LTE-001	Channel Bandwidth	5/10/15/20	MHz	
TX-LTE-002	Channel Offset	200	kHz	
TX-LTE-003	TxOFF residual noise	<-145	dBm/Hz	
TX-LTE-004	LTE EVM 256QAM	<3.5	%	
TX-LTE-005	LTE EVM 64QAM	<6	%	
TX-LTE-006	LTE EVM 16QAM	<10.5	%	
TX-LTE-007	LTE EVM QPSK	<14.5	%	
TX-LTE-008	Spurious (9KHz-150KHz)	<-36	dBm	1KHz Bandwidth
TX-LTE-009	Spurious (150KHz-30MHz)	<-36	dBm	10KHz Bandwidth
TX-LTE-010	Spurious (30MHz-1GHz)	<-36	dBm	100KHz Bandwidth
TX-LTE-011	Spurious (1GHz-12.5GHz)	<-30	dBm	1MHz Bandwidth
TX-LTE-012	Spectrum Emission mask	>3	dB Margin	Category B (Option 1)
TX-LTE-013	ACLR	>50	dBc	No need to be better than -18dBm/1MHz
TX-LTE-014	Time Alignment	<90	nS	
TX-LTE-015	Output Return Loss	>12	dB	

12.2. NR

Requirement	Parameter	Value	Unit	Comments
TX-NR-000	Number of Carriers	1		
TX-NR-001	Channel Bandwidth	5 to 100	MHz	
TX-NR-002	Channel Offset	200	kHz	
TX-NR-003	TxOFF residual noise	<-85	dBm/MHz	
TX-NR-004	LTE EVM 256QAM	<3.5	%	
TX-NR-005	LTE EVM 64QAM	<6	%	
TX-NR-006	LTE EVM 16QAM	<10.5	%	
TX-NR-007	LTE EVM QPSK	<14.5	%	
TX-NR-008	Spurious (9KHz-150KHz)	<-36	dBm	1KHz Bandwidth
TX-NR-009	Spurious (150KHz-30MHz)	<-36	dBm	10KHz Bandwidth
TX-NR-010	Spurious (30MHz-1GHz)	<-36	dBm	100KHz Bandwidth
TX-NR-011	Spurious (1GHz-12.5GHz)	<-30	dBm	1MHz Bandwidth
TX-NR-012	Spectrum Emission mask	>3	dB Margin	Category B (Option 1)
TX-NR-013	ACLR	>50	dBc	No need to be better than -15dBm/1MHz
TX-NR-014	Time Alignment	<65	nS	
TX-NR-015	Output Return Loss	>12	dB	

13. Receiver

Requirement	Parameter	Value	Unit	Comments
RX-GEN-000	Number of Carriers	2x LTE or 1x NR		
RX-GEN-001	Max. Input power	<-35dBm	dBm	Composite Power

13.1. LTE

Requirement	Parameter	Value	Unit	Comments
RX-LTE-000	Number of Carriers	1		
RX-LTE-001	Channel Bandwidth	5/10/15/20	MHz	
RX-LTE-002	Channel Offset	200	kHz	
RX-LTE-003	Reference Sensitivity 5MHz	<-103.5	dBm	
RX-LTE-004	Reference Sensitivity 10MHz	<-103.5	dBm	
RX-LTE-005	Reference Sensitivity 20MHz	<-103.5	dBm	
RX-LTE-006	Dynamic Sensitivity 5MHz	<-72.2	dBm	Interferer -82.5dBm
RX-LTE-007	Dynamic Sensitivity 10MHz	<-72.2	dBm	Interferer -79.5dBm
RX-LTE-008	Dynamic Sensitivity 20MHz	<-72.2	dBm	Interferer -79.5dBm
RX-LTE-009	E-UTRA Narrow band Adj. Channel Selectivity 5MHz[1]	Pref + 1.5dB	dBm	Interferer -49dBm
RX-LTE-010	E-UTRA Narrow band Adj. Channel Selectivity 10MHz[1]	Pref + 1.5dB	dBm	Interferer -49dBm
RX-LTE-011	E-UTRA Narrow band Adj. Channel Selectivity 20MHz[1]	Pref + 1.5dB	dBm	Interferer -49dBm
RX-LTE-012	E-UTRA Adj. Channel Selectivity 5MHz[1]	Pref + 1.5dB	dBm	Interferer -52dBm
RX_LTE-013	E-UTRA Adj. Channel Selectivity 10MHz[1]	Pref + 1.5dB	dBm	Interferer -52dBm
RX_LTE-014	E-UTRA Adj. Channel Selectivity 20MHz[1]	Pref + 1.5dB	dBm	Interferer -52dBm

Note 1. See Ref 1§7.5 for interferer characteristics

Requirement	Parameter	Value	Unit	Comments
RX-LTE-016	Generic Blocking[1]	-43	dBm	+20/-20MHz
RX-LTE-017	Generic Blocking[1]	-15	dBm	+CW further
RX-LTE-018	Intermodulation[2]	-52	dBm	

Note 1. See Ref 1§7.6.1 for interferer characteristics
Note 2. See Ref 1§7.8.1 for measurement characteristics

13.2. NR

Requirement	Parameter	Value	Unit	Comments
RX-NR-000	Number of Carriers	1		
RX-NR-001	Channel Bandwidth	10 to 100	MHz	
RX-NR-002	Channel Offset	200	kHz	
RX-NR-003	Reference Sensitivity 5/10/15MHz SBS 15KHz	<-103.7	dBm	In 25 RB -> 4.5MHz
RX-NR-004	Reference Sensitivity 10/15MHz SBS 30KHz	<-103.8	dBm	In 11 RB -> 4MHz
RX-NR-005	Reference Sensitivity 10/15MHz SBS 60KHz	<-101.9	dBm	In 11 RB -> 7.9MHz
RX-NR-006	Reference Sensitivity 20-50MHz SBS 15KHz	<-98.3	dBm	In 106RB -> 19.1MHz
RX-NR-007	Reference Sensitivity 20-100MHz SBS 30KHz	<-98.6	dBm	In 51RB -> 18.4MHz
RX-NR-008	Reference Sensitivity 20-100MHz SBS 60KHz	<-9.78	dBm	In 24RB -> 17.3MHz
RX-NR-009	Dynamic Sensitivity 5-100MHz	Ref 2§7.3.2-1		
RX-NR-010	E-UTRA Narrow band Adj. Channel Selectivity 5-100MHz[1]	Pref + 1.5dB	dBm	Interferer -49dBm
RX_NR-011	E-UTRA Adj. Channel Selectivity 5-100MHz[1]	Pref + 1.5dB	dBm	Interferer -52dBm

Note 2. See Ref 1§7.5 for interferer characteristics

Requirement	Parameter	Value	Unit	Comments
RX-NR-013	Generic Blocking[2]	-43	dBm	+20/-20MHz
RX-NR-014	Generic Blocking[2]	-15	dBm	+CW further
RX-NR-015	Intermodulation[3]	-52	dBm	

Note 1. See Ref 2§7.6.1 for interferer characteristics
Note 2. See Ref 2§7.8.1 for measurement characteristics

14. Control & Management

Requirement	Parameter
CMM-000	Control & Management based on ORI specification Ref. 15,16 & 17.
CMM-001	Supported Object: - RE - TxSigPath_EUTRA - RxSigPath_EUTRA - TxSigPath_NR - RxSigPath_NR - oriLink - antPort
CMM-002	Supported Device Management Request: - HealthCheck - set Time - RE Reset
CMM-003	Supported Software Management Request : - Version Query - Software Update Preparation - Software Download - Software Activation
CMM-004	Support Fault Management

15. Environmental

Requirement	Parameter	Value	Unit	Comments
ENV-000	Operating Temp. Range Min.	-40	°C	Ref. 5 Class4.1
ENV-001	Operating Temp. Range Max.	+55	°C	Ref. 5 Class4.1 (with sun cover)
ENV-002	Storage Temp. Range Min.	-40	°C	Ref. 3 Class2.3
ENV-003	Storage Temp. Range Max.	+70	°C	Ref. 3 Class2.3
ENV-004	Transportation Temp. Range Min.	-40	°C	Ref. 4 Class1.2
ENV-005	Transportation Temp. Range Max.	+70	°C	Ref. 4 Class1.2
ENV-006	Shall respect at least IP66 as defined by the document Ref. 9 when in approved operational condition			
ENV-007	Shall not resonate in audibled range (20Hz-20KHz) when in approved operational condition			

16. Regulatory

Requirement	Parameter
REG-000	Product shall pass and have the CE marking
REG-001	For electrical safety product shall comply to requirement defined per Ref. 10
REG-002	Product shall comply to requirement defined per Ref. 11 & 12
REG-003	ROHS : the product and its internal components shall have to fulfill the requirements Ref. 19
REG-004	WEEE : the product and its internal components shall have to fulfill the requirements of Ref. 20
REG-005	REACH: AW2S shall fulfill at any time all requirements according to the regulation Ref 21 concerning the handling of chemical substances. AW2S shall especially fulfill all duties imposed upon him according to Articles 31 to 33 (incl.) and shall provide all information which the customer may require.

- Swallow V6 -
LTEENB Transceiver User Guide

Reference :
DSG000017
Author :
M.ZEGHERS
Revision :
J

1 Document history

Date	Rev.	Author	Comments
18/10/2018	A	M.ZEGHERS	Initial release
19/11/2018	B	M.ZEGHERS	Updated for 6.4 release: - Added type D board (Lynx-R IB) support
28/01/2019	C	M.ZEGHERS	Updated for 6.5 release: - Updated Figure 22: Swallow configuration file content - Added `wait-mode` parameter to CPU management section.
05/02/2019	D	M.ZEGHERS	Updated for 6.6 release: - Added TRX API 14 support
02/09/2019	E	M.ZEGHERS	Updated for 6.9 release: - Added TRX API 15 support - Added type E board (Lynx-R-V2 IB) support - Added LTEUE support - Updated Features and Improvements over the V5 LTEENB Transceiver sections - Updated Software installation section - Added Built-In Self-Test (BIST) section - Added Radio Management Unit (RMU) section - Updated Swallow driver interface file (swallow.cfg) section for Multi-RAT support - Updated Swallow configuration file (swallow.xml) section for Multi-RAT and LTEUE support - Added section UE mode cells definition - Moved section PCIe and CPRI bandwidth into annexes - Moved section Example RAN setup into annexes
10/02/2020	F	M.ZEGHERS	Updated for 6.10 release: - Added 5G-NR support

			- Updated section Software requirements - Updated Features and Improvements over the V5 LTEENB Transceiver sections - Updated table Signal carrier bandwidth usage
09/02/2021	G	M.ZEGHERS	Updated for 6.11 release: - Added section Cell time offset - Added Annex C: Example multi-board LTEENB configuration
26/03/2021	H	M.ZEGHERS	Updated for 6.12 release: - Updated section Improvements over the V5 LTEENB Transceiver (for SNMP support) - Added section Compiling the Net-SNMP software suite - Updated section Radio Management Unit (RMU) for SNMP support and new CLI/Logging information - Added section User interface (SNMP/CLI)
10/01/2022	I	M.ZEGHERS	Updated for 6.13 release: - Added type F board (Leonardo IB) support - Added section Custom channel filtering
21/02/2022	J	M.ZEGHERS	Updated for 6.14 release: - Added support for `pps-rise` and `pps-fall` external synchronization modes.

2 Approvals

Date	Rev.	Reader(s)	Comments
18/10/2018	A	N.BREANT	

3 Table of contents

4 Scope of this document

This document describes the steps the user should follow to install, setup and operate the AW2S PCI-Express transceiver board (also called Swallow board) V6 as a RF driver for Amarisoft LTEENB software stack. Using the PCI-Express transceiver board allows AW2S Remote Radio Heads to be seamlessly interfaced with the LTEENB software and thus greatly extend the coverage of the eNodeB.

5 Requirements

5.1 Hardware requirements

An AW2S PCI-Express transceiver board (Swallow) of any hardware type and revision, either:

- Type A: SwallowV1 (PRB000058), or
- Type B: SwallowV1.5 (PRB000119), or
- Type C: SwallowV2 (PRB000116), or
- Type D: Lynx-R (IB), or
- Type E: Lynx-R-V2 (IB), or
- Type F: Leonardo (IB).

A computer with the following setup:

- A fairly recent Intel x86_64 CPU such as Intel Core i7 or Intel Xeon E5 v3 with AVX2 support.
- At least 2GB of RAM. Fast RAM is preferable.
- At least 1GB of disk space.
- At least 1 Gigabit Ethernet port.
- If using a type A or type B Swallow board:
  - One available PCIe 4-lanes Gen2 slot with room for a full-height, half-length card.
- If using a type C Swallow board:
  - One available PCIe 8-lanes Gen3 slot with room for a full-height, half-length card.

If GPS synchronization is required:

- An active GPS antenna with SMA connector, supporting a biasing voltage of 3.3V.

5.2 Software requirements

The host computer of the LTEENB software and PCI-Express board needs to run on a 64-bit Linux distribution. The following distributions and Linux kernel versions are known as compatible (other distributions may be compatible, without guarantee):

- Distributions: Ubuntu 18.04, Ubuntu 20.04, Fedora 26, Fedora 27, CentOS 7.
- Kernel versions: 3.9 to 4.14.

Installed Amarisoft LTEENB software with valid licenses (contact Amarisoft to obtain a license).

- Amarisoft LTEENB release using TRX API 12 to 15.
  - TRX API for a given Amarisoft release can be found in file trx_driver.h of the trx_example tarball inside the Amarisoft installation package.
  - Note: Amarisoft LTEENB version must be at least 2020-01-23 for correct 5G-NR TDD mode of operation.

Internet access during the installation may be needed to download required third-party packages.

6 PCI-Express board (Swallow) description

6.1 Features

The Swallow board is a PCI-Express card that implements a specialized interface between the host computer and the wireless telecom industry standard CPRI protocol. The board can be used to interface the software-running eNodeB (also known as REC for Radio Equipment Controller) to multiple Radio Heads (also known as RE for Radio Equipment) via fiber link, effectively forming the fronthaul portion of the Radio Access Network (RAN).

The board's highly flexible design makes it suitable for UMTS (3G), LTE (4G), LTE-A (4G+) and NR (5G) wireless technology standards. It supports both Frequency-Division Duplexing (FDD) and Time-Division Duplexing (TDD) transmission modes. The board also support multi-RAT, multi-process applications, custom frequency bands, inverted frequency bands and UE mode of operation.

Using optimized Direct Memory Access (DMA) bus mastering over the PCI subsystem, the board transfers baseband I/Q samples directly from the host computer's memory into the CPRI frames for Downlink direction, and vice-versa for Uplink direction. The board also offers Control and Management (C&M) side-band channels for RE configuration and monitoring, via Ethernet interfaces through the CPRI links.

The board can also optionally synchronize itself on diverse type of external synchronization signals (such as GPS) to provide time and frequency synchronization. This allows minimizing of interference between different adjacent cells and RAN nodes, which is almost mandatory when using TDD mode of operation.

The board features the following components:

- Up to 4 independent master CPRI interfaces via SFP connectors. For each CPRI interface:
  - For type A board: support of up to CPRI line bit rate option 6 (6144 Mbps).
  - For type B, C, D, E and F boards: support of up to CPRI line bit rate option 8 (10137.6 Mbps).
  - Fast C&M side-band channel (Ethernet) for control plane.
  - Flexible mapping of AxC containers (I/Q data) into the CPRI frames.
  - Automatically detected star or daisy-chained fronthaul topology configurations.
  - LEDs provide visual indication of link status.
- Up to 8 independent radio cells, allowing Carrier-Aggregation and/or Multi-Sector applications. For each radio cell:
  - Any sample rate from 3.84 MSps to 188.16 MSps, in multiples of 3.84 MSps. Suitable for:
    - 3G UMTS,
    - 4G LTE channel bandwidths (including oversampled NB-IoT),
    - 5G NR channel bandwidths.
  - Any I/Q component sample width up to 16 bits.
  - Optional I/Q compression (down-sampling and non-linear quantization).
  - Up to 8 Tx and Rx channels (carriers) per radio cell, allowing MIMO 8x8 applications. Each channel:
    - can be enabled independently,
    - can be mapped independently to any AxC container of any CPRI master interface,
    - provides means of compensating for fiber link and RE internal delays.
- External input synchronization SMA connector with embedded GPS receiver, supports either (software selectable):
  - Active GPS antenna, in which case the board synchronizes to the absolute time reference.
  - UMTS 10ms synchronization pulse, allows synchronization in adjacent cells.
  - External PPS pulse.

- 15.36 MHz reference clock.

❖ Buffered output synchronization U.FL connector.
- Forwards the signal received on the input SMA connector (except GPS RF signal).

❖ Output reference connector (software selectable).
- Regenerated UMTS 10ms synchronization pulse.
- 15.36 MHz reference clock.

❖ Internal PLL and Stratum-3 compliant crystal oscillator for very low jitter clocks that meet CPRI and LTE requirements.

Note: The number of available CPRI interfaces, the number of available radio cells, and the number of available channels per radio cell depends on the actual firmware loaded into the board's programmable logic. See section Firmware versions for firmware ordering information to determine the actual values.

6.2 Mechanical and connectors

The PCI-Express transceiver board (Swallow) is a PCI-Express card with full-height, half-length form factor.

The board's PCB top side and front panel views for hardware type A and type B are shown in Figure 1.

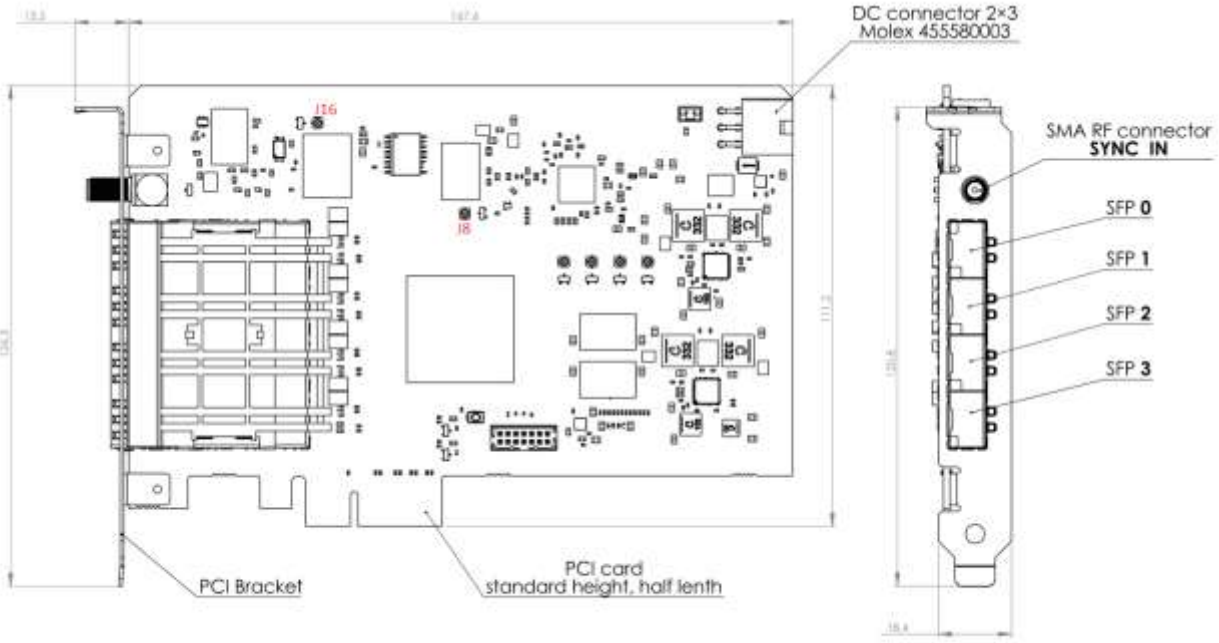


Figure 1: PCB top side and front panel views for type A and type B boards (dimensions in millimeters)

The board's PCB top side and front panel views for hardware type C are shown in Figure 2.

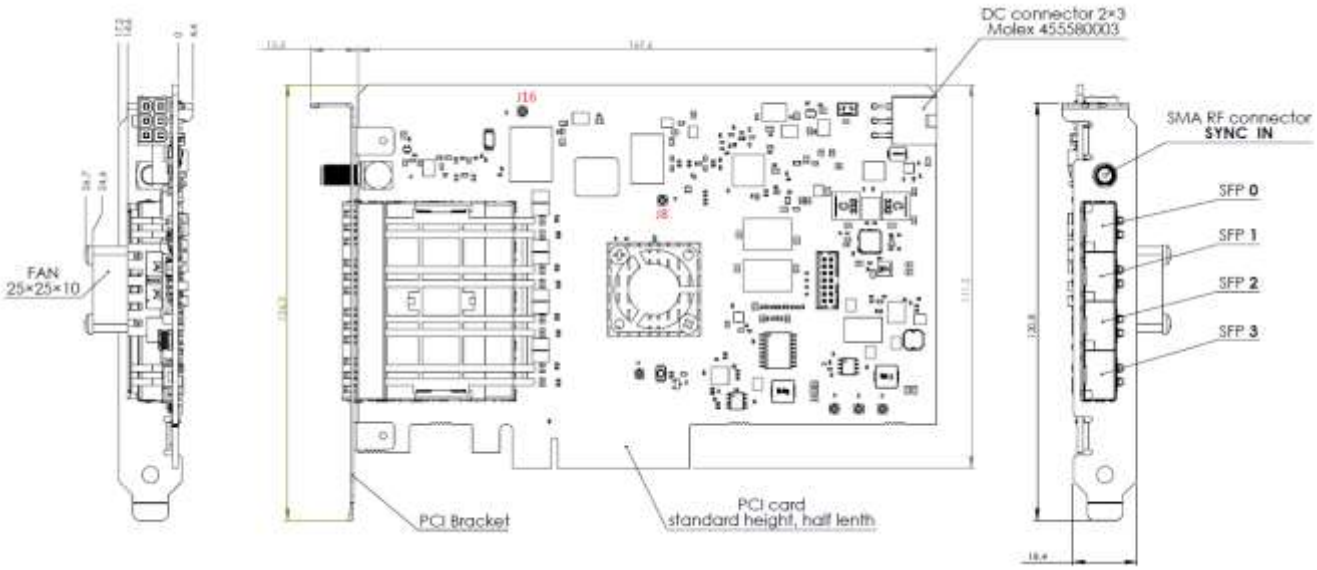


Figure 2: PCB top side and front panel views for type C board (dimensions in millimeters)

On the front panel there is one SMA input connector (SYNC_IN) which can be connected to either one of three types of external sources for synchronization:

- Active GPS antenna. Connector is biased to 3.3V when GPS synchronization is selected by software.
- LVCMOS 3.3V UMTS synchronization signal.
- LVCMOS 3.3V 15.36 MHz reference clock.

On the front panel there are 4 SFP connectors which can accept SFP copper cables or SFP transceivers for the fiber links. On each SFP connector, there are LEDs indicating the status of the CPRI link.

- Red LED indicates interface is activated but link is down (not operational).
- Green LED indicates interface is activated and link is up (operational).
- LED is turned off indicates interface is not available (not populated in firmware).

On the PCB top side, there is one output U.FL connector (J16) which is a buffered version of the input signal from the front panel SMA connector. There is also a second output U.FL connector (J8) which can be set via software to output either a 15.36 MHz reference clock or an internally regenerated UMTS synchronization signal.

6.3 Improvements over the V5 LTEENB Transceiver

The V6 version of the LTEENB Transceiver features massive improvements compared to the V5 version, without the need for a change in hardware. The following list is a non-exhaustive changelog of the upgrade from V5 to V6:

- Drastically reduced programmable logic footprint.
  - Possibility to use more CPRI interfaces, radio cells and data channels.
- CPRI line bit rate support increased from option 6 (6144 Mbps) to 8 (10137.6 Mbps), depending on board hardware type.
- Each radio cell data channel can now be mapped, independently, to any CPRI interface.
  - Possibility to extend MIMO capabilities by using more RRHs (i.e. it is possible to do MIMO 4x4 by using two 2x2-capable RRHs).
- CPRI daisy-chaining now supported.
- Carrier-Aggregation and sectorization now fully supported (on 1 or more RRHs).
- Maximum sample rate increased from 30.72 MSps to 188.16 MSps.
- I/Q compression now supported (including lossless compression) to reduce PCIe bandwidth, CPRI bandwidth and CPU usage.
- Improved delay compensation, all data channels can now be compensated independently.
- FDD and TDD modes of operations (including mixed TDD configurations) can now be run concurrently.
- Different sample rates (channel bandwidths) can now be mixed and run concurrently.
- Reduced GPS synchronization potential timing error from around 1µs to around 50ns.
- Major reduction of CPU usage.
- RRH Contol & Management can now be centralized in a Radio Management Unit software stack.
- Added multi-process capability, enabling the possibility to share a single Swallow board across multiple instances of the LTEENB and/or other software I/Q stacks.
- Added support for custom frequency bands, inverted frequency bands, and UE mode of operation (compatibility with LTEUE software).
- Added support for 5G-NR radio access technology.
- Multiple boards supported in a single computer.
- N-to-1 CPRI fronthaul mapping (transport I/Q samples across parallel CPRI ports to a single Radio Equipment).
- SNMP AgentX sub-agent implementation in the Radio Management Unit, allowing basic Radio Equipment lifecycle management and system monitoring.

7 Versioning and ordering information

There are three levels in versioning that needs tracking when using the Swallow board as transceiver:

- The board hardware type and revision,
- The firmware version that is programmed into the board's programmable logic,
- The software version that runs on the host computer.

7.1 Board hardware types and revisions

The board's hardware type and revision informs the state of the board's physical components and can play a role in the capabilities of the board. There are three hardware types of the board available:

- The type A board, also called SwallowV1 (PRB000058), is the entry-level version and can run CPRI line bit rate up to option 6 (6144 Mbps), using a 4-lanes Gen2 PCI-Express interface.
- The type B board, also called SwallowV1.5 (PRB000119), is an upgrade to the type A board. It provides the same functionalities but can run CPRI line bit rate up to option 8 (10137.6 Mbps).
- The type C board, also called SwallowV2 (PRB000116), is an upgrade to the type B board. It provides more programmable logic resources, can run CPRI line bit rate up to option 8 (10137.6 Mbps), and uses a 8-lanes Gen3 PCI-Express interface (quadrupling the PCI-Express bandwidth over type A and type B boards).
- The type D board, also called Lynx-R (IB), is equivalent to the type C board in terms of capabilities but is integrated inside the Lynx-R product.
- The type E board, also called Lynx-R-V2 (IB), is equivalent to the type C board in terms of capabilities but is integrated inside the Lynx-R-V2 product.
- The type F board, also called Leonardo (IB), is equivalent to the type C board in terms of capabilities but is integrated inside the Leonardo product.

7.2 Firmware versions

The firmware is defined as the configuration data (bitstream) loaded into the board's programmable logic (FPGA). Due to resource limitations of the programmable logic, the actual loaded firmware is the major decider of the capabilities of the board in terms of number of CPRI interfaces, radio cells and data channels available. Nevertheless, the key advantage of programmable logic is the possibility to be easily and quickly reconfigured as per system requirements. The reconfiguration process can be done via software on-site and remotely.

It should be noted that firmware versions are locked to specific hardware types. I.e. the configuration data is not the same for a type A board than it is for a type C board. However, there are equivalence in firmware versions between different hardware types.
The configuration data is available as a single `.bit` file with the ordering information explained in Figure 3.

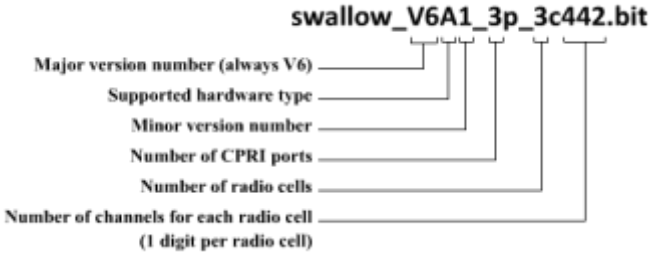


Figure 3: Firmware ordering information

The user should take into account that the number of CPRI ports, the number of radio cells, and the number of channels per radio cells all increase the usage of the (limited) resources of the programmable logic, so all combinations are not always feasible. Specific combinations can be tested for feasibility and made available upon request.

For instance if the system, using a type C board, requires only one 10Gbps CPRI port and only one radio cell but in MIMO 8x8 configuration, using a firmware such as `swallow_V6C1_1p_1c8.bit` would be a fitting choice. Another example would be if the system, using a type A board, requires four 6Gbps CPRI ports and 4 radio cells but in SISO configuration only, then a firmware such as `swallow_V6A1_4p_4c1111.bit` would most likely fit.

Refer to section Firmware reprogramming for instructions on how to reconfigure the board's programmable logic.

7.3 Software versions

The software is defined as the driver, libraries and applications used to operate the Swallow board on the host computer.

The software is installed on the computer using the Swallow Installer package, for which only the minor version may vary upon software upgrades. The full software is available from a single installer `.tar.gz` archive with the ordering information explained in Figure 4.

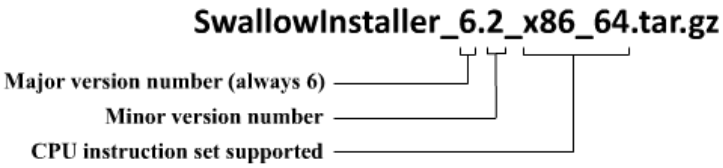


Figure 4: Software ordering information

While the major version number of the software and firmware versions must always match for correct operation, the minor versions are not correlated and need not be the same.

Note that some hardware types are only supported starting from specific software versions.

Refer to section Software installation for instructions on how to install, reinstall or uninstall the Swallow software suite.

8 Installation

8.1 Hardware setup

The hardware setup consists in installing the PCIe board into the computer and checking that it is recognized by the operating system.

1) Power down the computer and disconnect power supply to prevent any electrical damage to the board and the computer.
2) Plug the PCIe board into a compatible PCIe slot. It is preferable to plug the board into the closest slot from the CPU, for performance reasons (the least PCIe switches between the PCIe endpoint and the PCIe root port, the better).
3) Make sure that the PCI bracket is secured and stable.
4) It is not mandatory to connect a 12V power supply onto the DC connector of the PCIe board as power drawn from the PCIe slot is sufficient (less than 20 Watts).
5) Reconnect the computer power supply and power it up.
    a. Some LEDs on the board's PCB and the SFP ports should light up, indicating that the board correctly powered up and loaded its firmware from non-volatile memory.
6) Once the Linux OS has finished booting, use the command `sudo lspci -vv` and look for a PCI device named "`Non-VGA unclassified device: Xilinx Corporation Device`" or "`RF controller: Xilinx Corporation Device`".
    a. The output for this device should be similar to the one shown in Figure 5.
    b. Check that the line `LnkSta` shows the following values:
        i. For type A and type B boards: `Speed 5GT/s` and `Width x4`.
        ii. For type C, type D, type E and type F boards: `Speed 8GT/s` and `Width x8`.
    c. If the PCI device does not show or the link status is different from the one expected in previous step, refer to the Annex D: Troubleshooting section of this document.

Figure 5: lspci output for Swallow type A and B boards

8.2 Software installation

Software installation is the process of installing on to the host computer all software components required for the usage of the Swallow board with Amarisoft LTEENB software. The following components are to be installed:

- Swallow PCI-Express Linux driver kernel module.
- Environment auto-init and service scripts.
- Swallow library, Open Radio Interface client library, RAT C&M library, LTEENB TRX legacy and Multi-RAT driver libraries.
- DHCP server daemon and DHCPDUMP application.
- Extra applications such as `swallow_fwupgrade` for firmware reprogramming, `swallow_bist` as a Built-In Self-Test tool, `swallow_bwtest` for PCI-Express bandwidth measurements and `swallow_scan` for a simple uplink band scanning tool.
- Radio Management Unit software (`swallow_rmu`) optional application and service to provide centralized Remote Radio Heads Control & Management.

Installation requires root privileges on the host computer. Internet access might be needed to download third-party packages.
Extract the Swallow Installer package in any directory and enter the extracted archive, e.g.:

```
tar -xzvf SwallowInstaller_6.14_x86_64.tar.gz
cd SwallowInstaller_6.14_x86_64
```

Before installation, it is required to uninstall any other Swallow installation that may be present on the system beforehand, to do so:

```
sudo make uninstall
sudo make clean
```

If previous installation was done via the legacy installer (most likely when upgrading from older V5 releases), a computer reboot might be needed to complete the uninstallation procedure, the installer will print a warning (and refuse further installation) if that is the case.

The installation may require specific external packages to compile the driver kernel module and DHCP server. Depending on the running Linux distribution:

- Ubuntu:
  - PCAP development library
    - `sudo apt-get install libpcap-dev`
- Fedora and CentOS:
  - GCC compiler
    - `sudo yum install gcc`
  - Kernel exploded source tree
    - `sudo yum install kernel-devel-$(uname -r)`
  - PCAP development library
    - `sudo yum install libpcap-devel`

To proceed with the installation, build the sources against your system (this can take a couple minutes):

```
sudo make
```

Then install all the software components onto the system:

```
sudo make install
```

Note that the computer's network might be briefly disabled as the NetworkManager service will be restarted during installation.

Make sure that the build and install successfully completed. An example trace of a successful installation on Ubuntu 16.04 is shown in Figure 6.

```
cp data/apps/dhcpdump/dhcpdump /usr/local/bin/dhcpdump && chmod +x /usr/local/bin/dhcpdump && ln -sf /usr/local/bin/dhc
cp data/apps/extra/swallow_fwupgrade /usr/local/bin/swallow_fwupgrade && chmod +x /usr/local/bin/swallow_fwupgrade && l
cp data/apps/extra/swallow_bwtest /usr/local/bin/swallow_bwtest && chmod +x /usr/local/bin/swallow_bwtest && ln -sf /us
# Install service
cd data/service && ./install.sh
/sbin/init: unrecognized option '--version'
*****************************************
*** Installation completed successfully. ***
*****************************************
```

Figure 6: Example successful software installation on Ubuntu 16.04

The user can then check that the driver was correctly installed by having a look into the system log, via the following command:

```
dmesg |grep swallow
```

The first line should show the following text: "`swallow: registering driver V6.X`", indicating that the driver was registered into the operating system's kernel, and is ready to probe Swallow boards. If a board has been plugged into a PCI-Express slot at this time, the system log should show firmware and hardware information, and end with a "`swallow0: ready`" line indicating that this board is ready for usage, similarly to the system log shown in Figure 7.

```
[    2.925873] swallow: module verification failed: signature and/or required key missing - tainting kernel
[    2.926080] swallow: registering driver V6.3
[    2.926101] swallow: probing new board
[    2.926174] swallow0: firmware version 6A3 (type A.1 hardware, [0x1])
[    2.926174] swallow0: 3 cells, 3 ports
[    2.940205] swallow0: cell 0, 4 antennas, Tx size = 512kB, Rx size = 512kB
[    2.940415] swallow0: cell 1, 4 antennas, Tx size = 512kB, Rx size = 512kB
[    2.940671] swallow0: cell 2, 4 antennas, Tx size = 512kB, Rx size = 512kB
[    3.004128] swallow0: Ethernet over CPRI0 active (swa0p0) with MAC: 00:00:5e:fa:00:00
[    3.064394] swallow0: Ethernet over CPRI1 active (swa0p1) with MAC: 00:00:5e:fa:01:00
[    3.124404] swallow0: Ethernet over CPRI2 active (swa0p2) with MAC: 00:00:5e:fa:02:00
[    3.124507] swallow0: GPS NMEA data forwarded to /dev/ttySwallowGps0
[    3.124638] swallow0: ready
```

Figure 7: Driver system log for a probed board

Following the installation, two system services are started. The first service ("`swallow`") handles access rights, network interfaces and DHCP server for the probed boards. The second service ("`rmu`") handles the Radio Management Unit application. The driver and services are automatically (re-)started upon installation and after system reboots.
The state of both services can be retrieved with the following commands, as shown in Figure 8.

```
sudo service swallow status
sudo service rmu status
```

Stopping the "`swallow`" service will cause Swallow network interfaces to be removed and the DHCP server to stop. Stopping the "`rmu`" service will cause the Radio Management Unit application to quit.

Both services can be stopped and restarted at any time (even during operation), at the potential cost of temporary RF service interruption.

```
user@pc-prod:~$ sudo service swallow status
● swallow.service - Swallow systemd script
   Loaded: loaded (/lib/systemd/system/swallow.service; enabled; vendor preset: enabled)
   Active: active (running) since mer. 2018-10-17 15:28:25 CEST; 5min ago
 Main PID: 5781 (service_swallow)
   CGroup: /system.slice/swallow.service
           ├─5781 /bin/bash /usr/local/bin/service_swallow.sh
           └─5795 /usr/local/sbin/dhcpd -f -lf /var/db/dhcp.leases -cf /usr/local/etc/dhcpd.conf swa0p0 swa0p1 swa0p2

oct. 17 15:28:25 pc-prod service_swallow.sh[5781]: Listening on LPF/swa0p1/00:00:5e:fa:01:00/192.168.128.0/24
oct. 17 15:28:25 pc-prod service_swallow.sh[5781]: Sending on   LPF/swa0p1/00:00:5e:fa:01:00/192.168.128.0/24
oct. 17 15:28:25 pc-prod dhcpd[5795]: Sending on   LPF/swa0p1/00:00:5e:fa:01:00/192.168.128.0/24
oct. 17 15:28:25 pc-prod dhcpd[5795]: Listening on LPF/swa0p0/00:00:5e:fa:00:00/192.168.127.0/24
oct. 17 15:28:25 pc-prod service_swallow.sh[5781]: Listening on LPF/swa0p0/00:00:5e:fa:00:00/192.168.127.0/24
oct. 17 15:28:25 pc-prod service_swallow.sh[5781]: Sending on   LPF/swa0p0/00:00:5e:fa:00:00/192.168.127.0/24
oct. 17 15:28:25 pc-prod service_swallow.sh[5781]: Sending on   Socket/fallback/fallback-net
oct. 17 15:28:25 pc-prod dhcpd[5795]: Sending on   LPF/swa0p0/00:00:5e:fa:00:00/192.168.127.0/24
oct. 17 15:28:25 pc-prod dhcpd[5795]: Sending on   Socket/fallback/fallback-net
oct. 17 15:28:25 pc-prod dhcpd[5795]: Server starting service.
```

Figure 8: Services status

If a Swallow board is present, has been probed by the driver, and the "`swallow`" service is running, the `ifconfig` command shows that extra network interfaces (one for each Swallow CPRI port populated by firmware) have been registered into the system, named `swaXpY` (`X` being the board's minor identifier, and `Y` the index of the CPRI port), as can be seen in Figure 9.

```
swa0p0    Link encap:Ethernet  HWaddr 00:00:5e:fa:00:00
          inet addr:192.168.127.1  Bcast:192.168.127.255  Mask:255.255.255.0
          inet6 addr: fe80::200:5eff:fefa:0/64 Scope:Link
          UP BROADCAST RUNNING  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:176 (176.0 B)

swa0p1    Link encap:Ethernet  HWaddr 00:00:5e:fa:01:00
          inet addr:192.168.128.1  Bcast:192.168.128.255  Mask:255.255.255.0
          inet6 addr: fe80::200:5eff:fefa:100/64 Scope:Link
          UP BROADCAST RUNNING  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:180 (180.0 B)

swa0p2    Link encap:Ethernet  HWaddr 00:00:5e:fa:02:00
          inet addr:192.168.129.1  Bcast:192.168.129.255  Mask:255.255.255.0
          inet6 addr: fe80::200:5eff:fefa:200/64 Scope:Link
          UP BROADCAST RUNNING  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:180 (180.0 B)
```

Figure 9: Ethernet over CPRI network interfaces

8.2.1 Compiling the Net-SNMP software suite

The installation of the Net-SNMP software suite on the host system is mandatory for the execution of the optional Radio Management Unit **when SNMP option is enabled**. If SNMP support is not desired, then the installation can be skipped. See section Activating/deactivating the RMU for information on how to enable/disable RMU SNMP option.

In the Swallow V6 transceiver system, the SNMP agent (which is implemented by the Net-SNMP software) running on the host system offers all necessary external (within reach of the NMS) control & monitoring functionalities. Those functionalities are further described in the section User interface (SNMP/CLI) of this document.

Due to the possibly long compilation and installation time of the Net-SNMP software suite, it is not installed automatically via the Swallow V6 installer `Makefile`. The operator has to compile and install the correct version of Net-SNMP manually.

Fortunately, the suitable Net-SNMP source code is available in the Swallow V6 installer package, inside the `snmp` directory. Alternatively, the source code can be downloaded from the Net-SNMP website http://www.net-snmp.org/.
To proceed, extract the source code as follow:

```
tar -xzvf net-snmp-5.9.tar.gz
cd net-snmp-5.9
```

Then follow the instructions from the README and INSTALL files located in that folder. The installation instructions are also available on the Net-SNMP website at address http://www.net-snmp.org/docs/INSTALL.html.
A minimal install only need those few commands to be executed:

```
./configure
make
sudo make install
```

At this point, most SNMP applications (`snmpd`, `snmptrapd`, `snmpget`, `snmpset`…) and libraries (`libnetsnmp`, `libnetsnmpagent`) should be installed on the system. Note that it may be necessary to execute the command `sudo ldconfig` for the linker to reference the new Net-SNMP library.

The following step is configuring the SNMP service. In the Swallow V6 system, only the `snmpd` application needs to be running on the host during operation, this is usually done by activating the SNMP daemon service. The configuration of `snmpd` is out of scope of this document, as configuration details are specific to the requirements of the network administrator. Documentation on how to configure Access Control Setup, Trap Forwarding and other features implemented by `snmpd` is readily available on the Net-SNMP website or many other sources.

In the Swallow V6 system, the `snmpd` needs to be setup as an Agent X master, this is done by adding the line `master agentx` to `snmpd.conf` file:

That being said, a minimal and unsecure `snmpd.conf` file is available in the `snmp` directory of the Swallow V6 installer package. This configuration file can be used for initial testing of the system, such as checking whether the system monitoring is working, from a remote NMS.

Finally, the Swallow V6 system Management Information Base (MIB) file `AW2S-RMUv6-MIB.txt` is also available in the `snmp` directory. This file references all management parameters and is typically imported into the NMS.

8.3 Firmware reprogramming

Warning: Reprogramming the Swallow board firmware is a risky process in the way that the operation must not be cancelled and the computer must not be powered off or restarted during the reprogramming procedure. Failure to take these precautions may result in corruption of the board's flash memory and thus the board may not be able to boot anymore. In case of flash corruption, the board will have to be shipped back to AW2S for factory reprogramming.

In all cases, firmware reprogramming requires that a valid software suite (compatible with the currently running firmware version) has been installed and that the board has been probed by the driver and is ready to use, as it uses in-system programming tools. No extra hardware is required for reprogramming.

For reprogramming, the user can use the `swallow_fwupgrade` application that is automatically installed on the computer via the Software Installer package and is executable from anywhere.

If reprogramming to a V6 firmware, the configuration data is a `.bit` file. Execute the application with the wanted `.bit` file as argument, e.g.:
`swallow_fwupgrade swallow_V6B2_3p_3c444.bit`
If reprogramming to a V5 firmware, the configuration data is a `.mcs` file. Execute the application with the wanted `.mcs` file as argument, e.g.:
`swallow_fwupgrade swallow_V5_10.mcs`

Note: If upgrading from a V5 firmware to a V6 firmware, the user must use the `swallow_fwupgrade` application that comes from the Swallow Installer V5 package. In this case, upgrading to V6 firmware is only supported starting from Swallow software V5.16 (e.g. `SwallowInstaller_5.16_x86_64.tar.gz` package).

The firmware reprogramming process begins and must not be cancelled in any way until it has completed. An example log of successful reprogramming is shown in Figure 10.

```
user@pc-prod:~$ swallow_fwupgrade swallow_V6A3_3p_3c444.bit
Processing Swallow board firmware update with file swallow_V6A3_3p_3c444.bit (4976640 bytes, 76 sectors)
Erase flash...
Erase completed
Program flash...
Flash programmation successful !
```

Figure 10: Example successful firmware reprogramming

Be aware that there are dependencies between the board's hardware type and the firmware version (compatibility issues may arise), in this case the application will exit with an error text indicating the failure reason and the flash memory is unaltered. For instance, it is not possible to downgrade a type C board with a V5 firmware.

When using multiple boards in a single host, you can select which board to upgrade by adding the board identifier (board index number, range is `[0...num_boards[`), parameter at the end of the command line, e.g.:
`swallow_fwupgrade file.bit 0`, or
`swallow_fwupgrade file.bit 1`.

The computer will need to be power cycled (full cold restart) for the new firmware to be loaded by the board.

9 Built-In Self-Test (BIST)

Swallow software version 6.9 introduces a user-triggered Built-In Self-Test application that can be run to fully and automatically test the functionalities of the Swallow board inside the host system.

The `swallow_bist` application is automatically installed on the computer via the Software Installer package and is executable from anywhere. The application is executed as CLI (command line interface), requires root privileges and takes full control of the Swallow board during the test.

Other applications (including the Radio Management Unit) may not run simultaneously as the BIST. For this reason, the user should make sure that all other applications susceptible to use the Swallow board are stopped before running the BIST. Usually, this can be done by stopping the LTE and RMU services:

```
sudo service lte stop
sudo service rmu stop
```

Those services can be restarted after the BIST terminates.

It is generally preferable to run the BIST once when installing a new system to check that the board operates correctly and at full capacity within the host system. It may also be safe to run the BIST at least once after a Swallow software or firmware upgrade.

Nominally, for more extensive testing, the Swallow board's SFP ports should be looped back externally using an external SFP loopback (via either an SFP loopback adapter module, or fiber loopback).
If external SFP loopback is not possible, the BIST provides an option to enable internal PCS/PMA loopback. This setup is named "local testing". In that case, the actual hardware SFP ports, SFP modules and fiber cables are not tested.

The `swallow_bist` accepts multiple options to constrain the behavior of the automated testing. For those options that are not provided by the user via CLI, the application will assume some kind of default behavior, detailed below.
The following list provides the synopsis of the application and details the options.

Synopsis:

```
swallow_bist [options] [minor]
```

Options:

-h, --help
Print the command line interface helper and exit.

-v, --version
Print the application's version and exit.

-f, --file <*filename*>
Log file name to output the test results to. When set, test results (full console log) are also copied into the file <*filename*>. By default, test results are not stored into a file.

-l, --local
Enable local testing. Set this option when external SFP loopback is not possible.

--skip-bw
Skip PCIe bandwidth testing. Set this option to skip the high-priority bidirectional PCIe bandwidth measurements.

--skip-gps
Skip GPS synchronization test. This option should be set when GPS antenna is not connected or GPS synchronization testing is not desired.

--skip-ports
Skip CPRI ports tests. Set this option to skip CPRI ports line bit rate testing and Ethernet over CPRI bandwidth testing.

--skip-cells-dma
Skip cells DMA tests. Set this option to skip tests related to memory transfers between card and host.

--skip-cells-fh
Skip cells fronthaul tests. Set this option to skip tests related to data transfers across the CPRI links (fronthaul interface).

--skip-cells
Skip all cells tests. Equivalent to setting both --skip-cells-dma and --skip-cells-fh options.

-t, --timeout-gps *<timeout>*
Timeout of GPS synchronization test in seconds. Value must be between 1 and 3600. Default value is 120. Not applicable if --skip-gps option is set.

-s, --speed *<option>*
Select CPRI line speed option for single-rate tests such as cells DMA testing. Value must be a supported CPRI line speed option for the underlying Swallow board. Default value is 3.

-e, --exclude-speeds *<option1,option2,...>*
Exclude comma-separated CPRI line speed options from multi-rate tests such as CPRI ports tests and cells fronthaul tests. By default, no CPRI line speed option is excluded. Setting this option may be useful when the external SFP loopback module does not support specific line speeds.

-d, --data-length *<length>*
Length of data buffers (in ms) for transmission, reception and correlation. Value must be between 1 and 50. Default value is 10. Larger value causes more stringent testing at the cost of extended test duration.

-c, --compr-sigma *<sigma>*
Set the I/Q compression sigma parameter for tests using I/Q non-linear quantization. Default value is 4000.

--no-delay-comp

Test without delay compensation. Setting this option will disable delay-compensation testing from all datapath related tests.

Arguments:

[minor]

Swallow device minor identifier. This argument can be set to select which Swallow board to test when multiple boards are present on the host system. Default value is 0.

Example BIST commands below:

Command	Description
`sudo swallow_bist`	Run all tests, in external SFP loopback mode, with default options.
`sudo swallow_bist --skip-gps -e 1,7`	Run all tests, except GPS synchronization test, in external SFP loopback mode. Exclude CPRI line speed options 1 and 7 (for instance if those line speed options are not supported by external SFP loopback adapter).
`sudo swallow_bist -l -f results.txt`	Run all tests, in local mode (no need for external SFP loopback), and store results into file results.txt.

Table 1: Example swallow_bist commands

When executed, the BIST starts up by displaying device information such as versioning, supported CPRI line speed options and capabilities. BIST options and parameters are also displayed for easier understanding of the BIST scope. An example BIST startup with command `sudo swallow_bist -l --skip-gps` is shown in Figure 11.

```
####################################
######   Device information   ######
####################################
Type B (Swallow V1.5) FW:6B5 HWrev:0x00 (minor=0)
Software version 6.9
Maximum CPRI line bit rate: option 8 (10137.6 Mb/s)
Number of cells: 6 (antenna channels list: 2, 2, 2, 2, 2, 2)
Number of fronthaul ports: 3
Current FPGA temperature: 67 degC
***** Testing in port local loopback mode *****
***** Skipping GPS synchronization test *****
```

Figure 11: BIST device information

After startup, the BIST will run all non-skipped tests in a sequence.

Depending on options passed to the application, the number of radio cells and CPRI ports populated in firmware, the BIST may take more than ten minutes to complete. The sequence can be safely interrupted at any time by issuing Ctrl+C. Once finished, final results are displayed in a table, as shown in Figure 12.

```
##############################
######   Final results   ######
##############################
+---------------------------------+-------+--------+--------+--------+
|                       Test Name | Count | Passed | Failed | Status |
+---------------------------------+-------+--------+--------+--------+
|            PCI-Express bandwidth |     6 |      6 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
| Time & Frequency synchronization |     1 |      1 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|                      CPRI ports |    48 |     48 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 0 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 0 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 1 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 1 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 2 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 2 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 3 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 3 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 4 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 4 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|     Cell 5 Host memory interface |   144 |    144 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|       Cell 5 Fronthaul interface |   288 |    288 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
|                           TOTAL |  2647 |   2647 |      0 |   PASS |
+---------------------------------+-------+--------+--------+--------+
BIST duration: 5 minutes and 50 seconds
```

Figure 12: BIST final results

The BIST is considered “PASS” when there is no failed check amongst all tests.

10 Radio Management Unit (RMU)

The Radio Management Unit (abbreviated RMU), is an optional application and service introduced in Swallow software version 6.7 with the aim to centralize all the Remote Radio Heads Control & Management links. By doing so, the C&M links are kept separate from the I/Q data planes (TRX drivers). The RMU automatically and efficiently manages the shared resources for optimal CPRI and Radio Equipment configuration, unlocking the multi-RAT and multi-process capabilities of the Swallow board.

Additionally, the RMU implements the control layer core functionalities of the system's front-haul interface, with an extensive command line interface and optional SNMP AgentX subagent implementation.

In order to maintain backward compatibility with previous Swallow software versions, the RMU is not activated by default on a fresh install. If the user wishes to use the RMU as the central C&M unit, then it needs to be manually activated as described in section Activating/deactivating the RMU.

10.1 Overview

A RAT stack, being defined as any software I/Q stack such as Amarisoft's LTEENB stack, needs some sort of mean to setup the Radio Equipment as required by its own configuration. Prior to the introduction of the RMU, each RAT stack would open a C&M channel with the Radio Equipment to fully manage it. However, Radio Equipment typically support only a single open C&M channel at any given time, thus preventing the aggregation of multiple RATs on a single Radio Equipment.

To break this limitation, the RMU decouples the C&M link between the RAT stack and the Radio Equipment from the actual RAT stack software. The RMU handles the Radio Equipment configuration through the usual Open Radio Interface (ORI), and is interfaced to the multiple RAT stacks via RAT C&M channels provided by the RAT C&M shared library included in the Swallow Software package.

The RMU service is expected to run continuously on the system, whether or not any RAT stack is currently running. Radio Equipment's are thus continuously monitored. At any time a RAT stack is executed on the system, it would issue, through the RAT C&M channels, a signal path configuration request to the RMU. The RMU then configures the Radio Equipment as required by the RAT stack, and informs the RAT stack how the signal path should be mapped across the front-haul shared resources.

The RMU, being capable of managing multiple RAT stacks, multiple Radio Equipment and even multiple Swallow boards, can keep track of (and optimize) all the shared resources and thus make sure that there is no resource conflict. It is also used to handle system-wide parameters, such as the hardware clock reference (e.g. GPS) and CPRI ports line speed setup, faults on the radio end, and more.

In any case the RMU or Radio Equipment is to be restarted or stopped while RAT stacks are operating, the RAT C&M shared library would automatically inform the RAT stacks that the C&M channels and/or Radio Equipment are not operational any more. The RMU or Radio Equipment can then be restarted, and operation resumed, without having to restart any RAT stack. Reversely, RAT stacks can be restarted and reconfigured independently of each other, whilst keeping the RMU running.

To synthesize, The RMU implements the following functions:

- Provide system level monitoring, such as alarms and statuses of the different subsystems.
- Provide extensive logging for the system, Radio Equipments (configuration, errors, alarms…) as well as RAT stacks services.
- Provide dynamic reconfiguration and monitoring capabilities by implementing a SNMP AgentX subagent, and command line interface options.
- Provide service to the RAT stacks with the use of RAT C&M library C&M links.
- Configure the Swallow board's system-level parameters.
- Handle C&M links with the Radio Equipment by parsing DHCP requests/responses from the DHCP server, opening and managing ORI connections.
- Create, configure and delete signal paths on Radio Equipment as required by the RAT stacks configurations, and send back configuration responses to those RAT stacks.
- Provide commands for special features such as file transfers via FTP.
- Handle the critical front-haul link (CPRI I/Q data plane) shared resource efficiently, with the support of star and daisy-chained CPRI topologies.

Figure 13 describes how the RMU is implemented within the host's software layers.

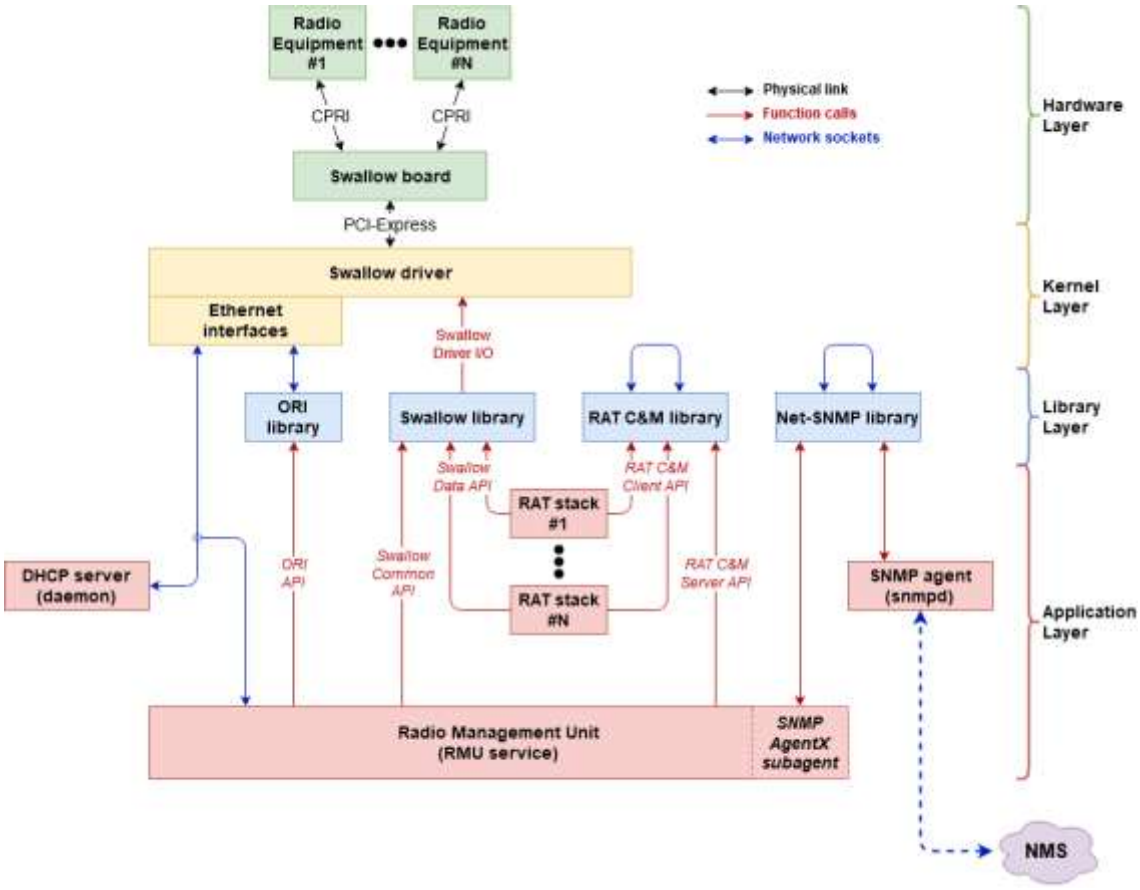


Figure 13: Radio Management Unit system diagram

10.2 Activating/deactivating the RMU

As described earlier, on a fresh Swallow software installation, the RMU service is not activated by default. It is the user's responsibility to manually activate the RMU if desired.

To do so, the Swallow software installer copies scripts into `/etc/rmu` folder, namely:
Execute `/etc/rmu/activate.sh` to activate the RMU (without SNMP support).
Execute `/etc/rmu/activate_with_snmp.sh` to activate the RMU (with SNMP support).
Execute `/etc/rmu/deactivate.sh` to deactivate the RMU.
These scripts provide non-volatile configuration of RMU activation, meaning that they are typically a one-time usage. Activation/deactivation persists after reboots and after Swallow software reinstallations.

It should be understood that the RMU service is always started at system reboot, but the actual RMU application (`swallow_rmu`) is executed by the service only when the RMU has been activated by above script.
Also note that when activating the RMU, the service is automatically restarted. In the other hand, when deactivating, the service is automatically stopped.

It is possible, when the RMU is activated, to manually start and stop the RMU service and application using the following commands:

```
sudo service rmu start
sudo service rmu stop
```

Note that starting the RMU service when the RMU is deactivated will not cause the RMU application to be executed.

Also, when the RMU is deactivated, it is possible to completely bypass the RMU service and launch the RMU application (`swallow_rmu`) manually if desired, using command line arguments as described in the next section.

Important: when using the RMU, the LTEENB RF driver must be the Swallow Multi-RAT transceiver (acting as a RAT stack, or RAT C&M client). See section Swallow driver interface file (swallow.cfg) for more information.

10.3 RMU application arguments

The `swallow_rmu` is the RMU's executable that is typically automatically executed by the RMU service, but may also be manually ran by the user.

The `swallow_rmu` accepts multiple options to constrain its behavior (mostly related to logging and SNMP). For those options that are not provided by the user via CLI, the application will assume some kind of default behavior, detailed below.
The following list provides the synopsis of the application and details the options.

Synopsis:

```
swallow_rmu [options] [path/to/config.xml]
```

Options:

```
-h, --help
     Print the command line interface helper and exit.
```

-v, --version
Print the application's version and exit.

--ld, --log-dir *<directory>*
Set log files working directory to *<directory>*. Default directory is /var/log/rmu. This is the directory where new log files are written.

--lf, --log-file *<filename>*
Specifies log file name. If not set, then the RMU generates a file name based on RMU start date and time. RMU-generated file names also provide file wrapping capability where a new file is created (with some index number in the resulting file name) once the file size reaches the value set by --log-size option.

--ls, --log-size *<size>*
Specifies log file maximum size in MBytes (applicable only if --lf option is not specified). Default is 0 (unlimited).

-c, --colored
When set, use ANSI coloring for console and log outputs, based on message severity.

-m, --mute
When set, mute console outputs.

-d, --debug
When set, output debug logs to console.

--snmp
Enable SNMP AgentX sub-agent.

-x *<agentxsocket>*
Sets the SNMP AgentX socket address. Equivalent to the -x option of SNMPD agent application (see snmpd manual).

-D
Disable most SNMP traps during the RMU init and exit processes.

Arguments:
[path/to/config.xml]
XML configuration file path. Default file path is /etc/rmu/rmu.xml.
The content and default values (in the case this argument is not specified) are defined in section RMU configuration file (rmu.xml).

Note that when the RMU application is executed via the RMU service, the following command line is invoked (as declared by RMU_ARGS variable in /usr/local/bin/service_rmu.sh script):
swallow_rmu /etc/rmu/rmu.xml --ls 5 -c -D

This means that the RMU service uses the XML configuration file `/etc/rmu/rmu.xml`, console and log outputs are colored, log files are output to `/var/log/rmu` with a maximum file size of 5 MBytes, and most init/exit SNMP traps are disabled.

10.4 RMU configuration file (rmu.xml)

The RMU configuration file (`rmu.xml`) is a XML file that includes all parameters than can be edited to configure the default behavior of the Radio Management Unit application.

When installing the Swallow software, a default `rmu.xml` file is copied in the `/etc/rmu` directory. Its default content is shown in below Figure 14.

rmu.xml ×

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Radio Management Unit configuration file -->

<rmu>

        <!-- Swallow configuration -->
        <swallow minor="*">
                <sync mode="gps" output="umts"/>
                <port id="*" line-speed="3" defragment="true"/>
        </swallow>

        <!-- Optional FTP links -->
        <ftp>
                <link alias="Releases">
                        <host>127.0.0.1</host>
                        <user>user</user>
                        <password>default0</password>
                        <directory>releases</directory>
                </link>
        </ftp>

</rmu>
```

Figure 14: RMU configuration file content

This file should typically be located in the `/etc/rmu` directory as this is the default location for the RMU service. However, nothing prevents the user from using a different path or file name should the RMU be executed manually.

The following sections provide details on how this file can be modified, with description of the different parameters, their acceptable values and admissible ranges. Note that when changing parameters in the XML file, the RMU application will need to be restarted (for instance by using `sudo service rmu restart`) for changes to take effect.

10.4.1 RMU Swallow configuration

Element:
`rmu -> swallow`

Cardinality:
Any.

Description:
This element is the parent element for one or multiple Swallow board's parameterizing. Actual configuration for the relevant Swallow board is deferred to the child elements, described below.

Parameter	Type	Description
`minor`	Unsigned integer or Wildcard	Selects to which Swallow boards this element refers to (indexed from zero). If `minor` is a wildcard "*", then this element refers to all Swallow boards. *Note: wildcarded elements are always parsed first by the application, then the numbered elements (gives priority to numbered elements).*

Table 2: RMU Swallow configuration parameters

10.4.2 RMU Swallow synchronization scheme

Element:
`rmu -> swallow -> sync`

Cardinality:
Unique.

Description:
This element allows configuration of the time and frequency synchronization schemes used and output by the PCIe board. An external time-synchronization source is highly recommended to operate and prevent interferences with adjacent radio nodes (especially in TDD transmission mode).

Parameter	Type	Description
`mode`	Enumerated string	Selects the synchronization source used by the board from the following values: `freerun`: The board runs on its internal clock oscillator and is not synchronized to any external source. `refclk`: The board synchronizes itself on an external 15.36 MHz reference clock from the SYNC_IN connector, providing frequency synchronization only. Invalid for type D, E and F boards. `gps`: The board synchronizes itself on the GPS system. Requires the usage of a GPS antenna on the SYNC_IN connector, providing time and frequency synchronization. `umts`: The board synchronizes itself on an external 100Hz UMTS synchronization pulse (e.g. from another Swallow board) on the SYNC_IN connector, providing time and frequency synchronization. Invalid for type D, E and F boards. `pps-rise`: The board synchronizes itself on the rising edge of an external PPS synchronization pulse on the SYNC_IN connector, providing time and frequency synchronization. Invalid for type D, E and F boards. `pps-fall`: The board synchronizes itself on the falling edge of an external PPS synchronization pulse on the SYNC_IN

		connector, providing time and frequency synchronization. Invalid for type D, E and F boards. Default value: `gps`
`output`	Enumerated string	Selects the synchronization signal type output from the board (not applicable to type D, E and F boards) that can be used to synchronize other nodes, from the following values: `reflck`: A 15.36 MHz clock is output on the J8 connector, providing frequency synchronization only to another board. `umts`: A 100Hz UMTS synchronization pulse is output on the J8 connector, providing time and frequency synchronization to another board. Default value: `umts`

Table 3: RMU Swallow synchronization scheme parameters

10.4.3 RMU Swallow port configuration

Element:
`rmu -> swallow -> port`

Cardinality:
Any.

Description:
This element provides Swallow CPRI port specific configuration.

Parameter	Type	Description
`id`	Unsigned integer or Widcard	Selects which CPRI port this element refers to (indexed from zero). If `minor` is a wildcard "`*`", then this element refers to all CPRI ports of the parent Swallow board. For instance, a value of 2 means that this element refers to the board's third CPRI port (i.e. CPRI interface on SFP_2 connector). *Note: wildcarded elements are always parsed first by the application, then the numbered elements (gives priority to numbered elements).*

`line-speed`	Unsigned integer	Sets the CPRI line bit rate option to be used on the selected Swallow CPRI port. This line bit rate option is also configured on all Radio Equipment CPRI master ports connected to this Swallow port (relevant only in daisy-chaining topology cases). Range (type A board): 2 to 6 Range (type B, C, D, E and F boards): 2 to 8 Default value: 3 *Note 1: Range may be further restricted by the line bit rate options supported by Radio Equipment and used SFP transceiver.* *Note 2: Due to different encoding, it is not possible to mix line bit rate option 8 with other line bit rates on the same board.*
`defragment`	Boolean	Can be either `true` or `false`. When a new RAT C&M link is serviced by the RMU, the allocation of signal paths on the CPRI links requires the usage of part of the CPRI basic frame for I/Q data plane transport. When RAT C&M links come and go, the CPRI basic frame may get fragmented over time and the need to defragment may be needed to allow further CPRI basic frame signal path allocation. However, the defragmentation of the CPRI basic frame implies short service interruption as position of the AxC containers may get shuffled and the Radio Equipment shall stop emission/reception for this period of time. For this reason, this parameter provides the possibility to enable or disable automatic defragmentation based on system requirements. When set to `true`, the RMU will defragment the CPRI basic frame if it detects that it is fragmented and a defragmentation may allow the allocation of more signal paths. This configuration guarantees optimal CPRI basic frame usage.

		However, service may be interrupted when a new RAT C&M link is created/removed. When set to `false`, the RMU never defragments the CPRI basic frame. This setting guarantees no service interruption, but the allocator may fail to allocate a signal path if the CPRI basic frame is fragmented. Default value: `true` *Note: To maintain backward compatibility with beta versions, the parameter values "always" and "on-demand" are interpreted as "true", and the value "never" is interpreted as "false".*

Table 4: RMU Swallow port configuration parameters

10.4.4 RMU FTP links

Element:
`rmu -> ftp -> link`

Cardinality:
Any.

Description:
This element defines a FTP link object, which facilitates file transfers to Radio Equipments using FTP, by providing FTP server information to the Radio Equipment for any file transfer procedure.
A FTP link object is merely a reference to a FTP server site, which does not reside inside the RMU. The actual FTP server can be installed separately on the host computer or any other computer available on the network that can be reached by Radio Equipments. It includes the FTP server host address, login credential and files directory linkage.

Parameter	Type	Description
`alias`	String (`AliasString`)	The `alias` attribute is a textual alias for the FTP link usable posteriorly to reference it, it does not affect functionality in any way and can be chosen arbitrarily. The textual convention for an `AliasString` is described in section Textual conventions. By default, no FTP link is defined. If any of the child parameters for a given link is missing or has a wrong value, then that link definition is ignored.

Table 5: RMU FTP link configuration parameters

Sub-element	Value type	Description
`host`	String (`IpAddress`)	IPv4 address (in dotted decimal notation) of the machine on which the FTP server is hosted. If the FTP server is hosted locally, then the loopback address 127.0.0.1 may be used.
`user`	String (`CredentialString`)	User identifier required for logging into the FTP server. The textual convention for a `CredentialString` is described in section Textual conventions.
`password`	String (`CredentialString`)	Password required for logging into the FTP server. The textual convention for a `CredentialString` is described in section Textual conventions.
`directory`	String (`DirNameString`)	FTP server working directory for this FTP link. Files transferred using this FTP link as reference will be located in this directory on the FTP server. The value `.` means that the working directory is at the root of the FTP server directory structure. The textual convention for a `DirNameString` is described in section Textual conventions.

Table 6: RMU FTP link configuration values

10.5 RMU console command line interface (CLI)

When the RMU service is running, it is possible to access the RMU console interface using the `screen` utility (requires `GNU screen` package to be installed on the system). Using the command `sudo screen -x rmu`, the user can gain access to the RMU console interface at any time. The screen session can be exited (without stopping the application), by issuing the *`Ctrl+A-D`* keyboard strokes. Refer to `GNU screen` documentation for more information on how to use this tool.

The RMU console interface, as shown in Figure 15, outputs logs in real-time and provides some commands to the user.

Logs are displayed with timestamp, log severity and, if applicable, the source object that emitted the log entry. By default (if `-d` option is not passed as argument to the application), all logs except debug logs are displayed in the console. If colored outputs are enabled, debug messages are displayed in blue, informative messages in green, warnings in yellow and errors in red colors.
Note that all logs displayed to the console are also copied into the RMU log files. Debug logs are always copied into the RMU log files (even when they are not being displayed in the console).

```
#
# cd /usr/local/bin
# ./swallow_rmu /etc/rmu/rmu.xml --ls 5 -c -D --snmp
@ 15:50:40.000 2021-03-23
        +0.234  info Starting up SNMP AgentX sub-agent
NET-SNMP version 5.9 AgentX subagent connected
        +0.262  info [rmu] Radio Management Unit v6.11
        +0.263  info [log] Status: Operational
        +0.263  info [cfg] Using /etc/rmu/rmu.xml config file
        +0.264  info [cfg] Status: Operational
        +0.264  info [cm] Status: Operational
        +0.265  info [swa:0] Type B (Swallow V1.5) FW6.7 (V6B7_3p_3c444.bit)
        +0.315  info [swa:0] Status: Pre-Operational
                                     Sync mode GPS: GPS/PPS missing
        +0.317  info [swa:0/port:0] Status: Pre-Operational
                                            CPRI link down
        +0.318  info [swa:0/port:1] Status: Pre-Operational
                                            CPRI link down
        +0.319  info [swa:0/port:2] Status: Pre-Operational
                                            CPRI link down
        +0.319  info [ftp] Creating FTP link [link:0] (Releases)
@ 15:50:41.000 2021-03-23
        +0.265  info [gps] Status: Pre-Operational
                                   No valid fix data available at the moment
        +0.266  info [swa:0/port:0/dhcp] Status: Operational
        +0.267  info [swa:0/port:1/dhcp] Status: Operational
        +0.268  info [swa:0/port:2/dhcp] Status: Operational
rmu>
```

Figure 15: RMU console interface example startup

Log entries are generally issued by objects which use a specific nomenclature that helps identifying the source of the information. Figure 16 below shows a hierarchical view of all the objects that are accessible via the console interface, or that may be identified in logs. Some objects may or may not be present at any given time, depending on hardware connections, current services or configuration. Other object types (such as `ftp/link:X`) may appear in multiple, distinct, instances. Table 7 provides a short description of each object type.

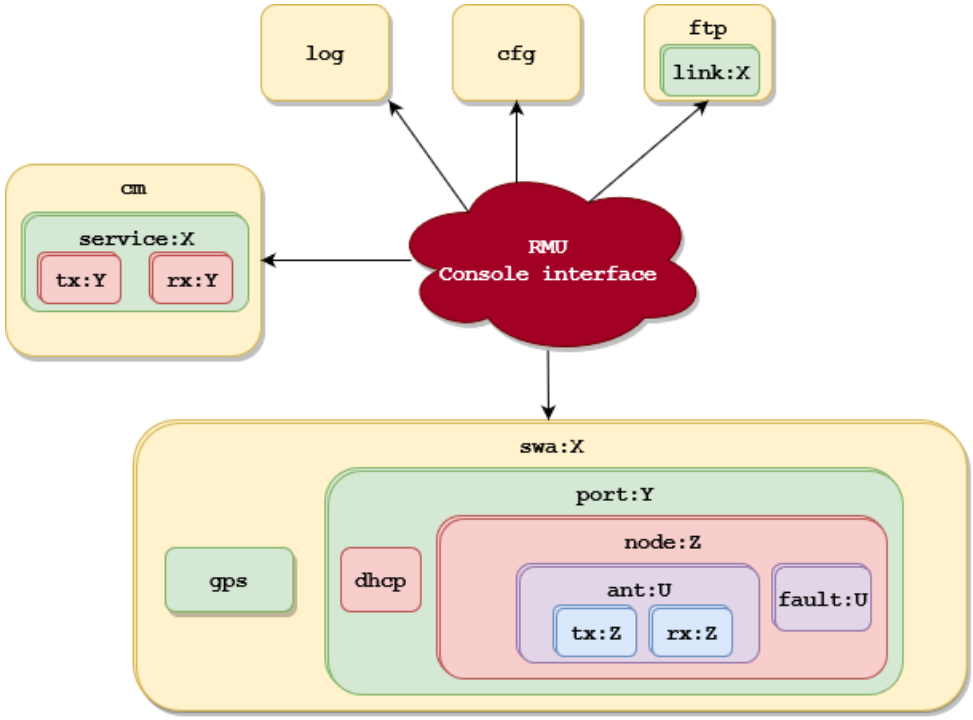


Figure 16: RMU console object diagram

Object	Description
`log`	Log and console manager: controls how logs are displayed and stored.
`cfg`	Configuration manager: handles non-volatile RMU configuration parameters.
`cm`	RAT C&M server: provides services to RAT stacks (RAT C&M clients).
`cm/service:X`	RAT C&M client service.
`cm/service:X/tx:Y`	RAT C&M TX signal path configuration request from a RAT C&M client.
`cm/service:X/rx:Y`	RAT C&M RX signal path configuration request from a RAT C&M client.
`swa:X`	Swallow board manager.
`swa:X/gps`	Swallow GPS NMEA parser: parses GPS fix & date information.
`swa:X/port:Y`	Swallow CPRI master port controller.
`swa:X/port:Y/dhcp`	DHCP parser on a Swallow CPRI master port `swa:X/port:Y`: parses incoming DHCP requests and extracts ORI-related parameters.
`swa:X/port:Y/node:Z`	Node (Radio Equipment) attached to Swallow CPRI master port `swa:X/port:Y`, using `Z` hops (node index in the daisy-chain list).

swa:X/port:Y/node:Z /fault:U	Active fault occurring on node swa:X/port:Y/node:Z. Note that the index U for a fault object is not an integer but a short string representing the fault type.
swa:X/port:Y/node:Z /ant:U	Antenna port index U on node swa:X/port:Y/node:Z
swa:X/port:Y/node:Z /ant:U/tx:V	TX signal path allocated on node antenna port swa:X/port:Y/node:Z/ant:U
swa:X/port:Y/node:Z /ant:U/rx:V	RX signal path allocated on node antenna port swa:X/port:Y/node:Z/ant:U
ftp	FTP manager: handles the list of FTP links defined in the RMU.
ftp/link:X	FTP link definition.

Table 7: RMU object nomenclature

The RMU console interface offers a range of commands that can be used to monitor objects, edit parameters, or trigger special operations. Basically, each object can be addressed individually, with most of them offering a list of commands specific to that object. Some commands also offer a recursive option, which means that it also applies to descendant objects.

The following commands are available for all objects, if <object> is empty, then the command targets the RMU application and no specific object:

<object> help [-r]

Prints the command helper for <object>, the helper includes the list of commands that can be used for <object>.
If –r option is set, then the call is recursive and command helpers are shown for all child objects of <object>.

<object> info [-r][-s][-v]

Print information for <object>.
If –r option is set, then the call is recursive and information is displayed for all child objects of <object>.
If –s option is set, only the object's operational statuses are shown.
If –v option is set, the displayed output is more verbose.

To exit the RMU application, the following RMU-level command is available:

exit

Exit the application. Note that if the RMU service is running, the application will automatically be restarted by the service. Alternatively, Ctrl+C can be used.

The list of commands for the different objects, and their effect, are explained with more details in section User interface (SNMP/CLI) of this document.

For example purpose, the sequence of commands displayed in Figure 17 shows how to disable a CPRI master port object: The list of commands for the target object is shown by using the `help` command, this helps finding out the command that should be used to change the Swallow CPRI master port enablement (among other possible commands). In this particular case, the CPRI master port is disabled by setting its line speed to the value 0.

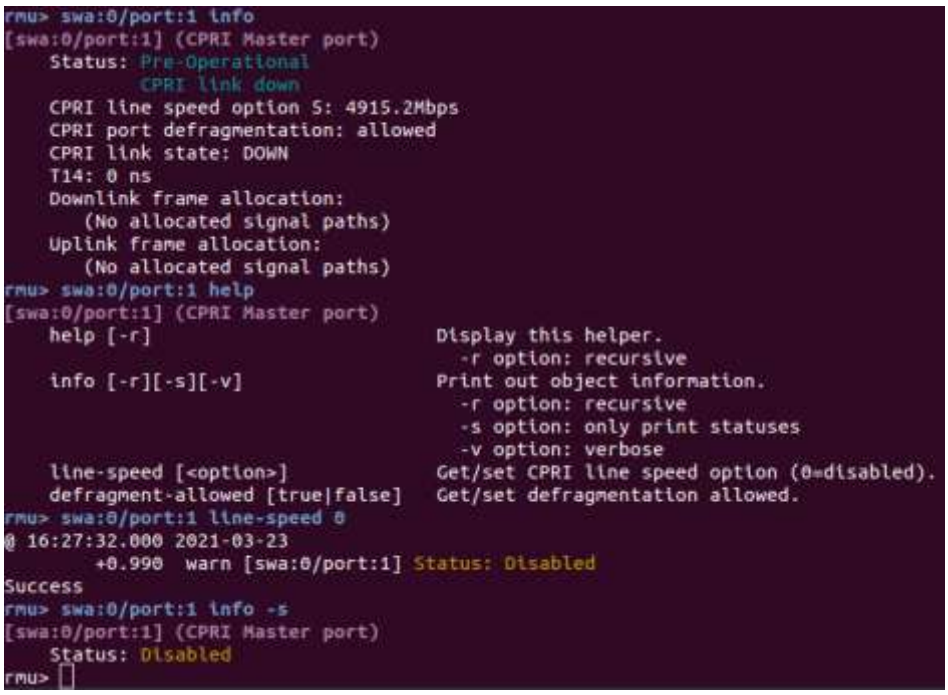


Figure 17: Example CLI sequence of commands

Note: Auto-completion is available for object completion, command completion and sometimes argument completion. Auto-completion is activated by hitting <TAB> keyboard key.

A particularly useful command is the top-level recursive status information command `info -rs`, this command provides a quick and clean overview of the statuses of all objects in the RMU, the output can be seen in Figure 18.

```
rmu> info -rs
Radio Management Unit
|
+---[log] (Log and Console Manager)
|       Status: Operational
|
+---[cfg] (Configuration Manager)
|       Status: Operational
|
+---[cm] (RAT C&M Manager)
|   |   Status: Operational
|   |
|   +---[cm/service:0] (OAI.27190.TRX0)
|       |   Status: Operational
|       |
|       +---[cm/service:0/tx:0] (TX NR 40MHz TDD @ arfcn=663334 / 30.00dBm)
|       |       Status: Errored
|       |               Target radio equipment not connected
|       |
|       +---[cm/service:0/rx:0] (RX NR 40MHz TDD @ arfcn=663334)
|               Status: Errored
|                       Target radio equipment not connected
|
+---[swa:0] (Swallow Manager)
|   |   Status: Pre-Operational
|   |           Sync mode GPS: GPS/PPS missing
|   |
|   +---[gps] (GPS NMEA Parser)
|   |       Status: Pre-Operational
|   |               No valid fix data available at the moment
|   |
|   +---[swa:0/port:0] (CPRI Master port)
|   |   |   Status: Pre-Operational
|   |   |           CPRI link down
|   |   |
|   |   +---[swa:0/port:0/dhcp] (DHCP Parser)
|   |           Status: Operational
|   |
|   +---[swa:0/port:1] (CPRI Master port)
|   |   |   Status: Disabled
|   |   |
|   |   +---[swa:0/port:1/dhcp] (DHCP Parser)
|   |           Status: Operational
|   |
|   +---[swa:0/port:2] (CPRI Master port)
|       |   Status: Pre-Operational
|       |           CPRI link down
|       |
|       +---[swa:0/port:2/dhcp] (DHCP Parser)
|               Status: Operational
|
+---[ftp] (FTP Manager)
    |
    +---[ftp/link:0] (Releases)
rmu>
```

Figure 18: Recursive status information command output extract

11 LTEENB/LTEUE transceiver configuration

This configuration section details how the Swallow board can be configured, after installation, for usage with Amarisoft LTEENB/LTEUE software as the "RF driver".

In addition to the Amarisoft LTEENB/LTEUE standard configuration file (e.g. `enb.cfg`, `gnb.cfg` or `ue.cfg`), the Swallow software requires two extra files (e.g. `swallow.cfg` and `swallow.xml`) to interface with LTEENB/LTEUE software. These three files are interlinked as explained in Figure 19.

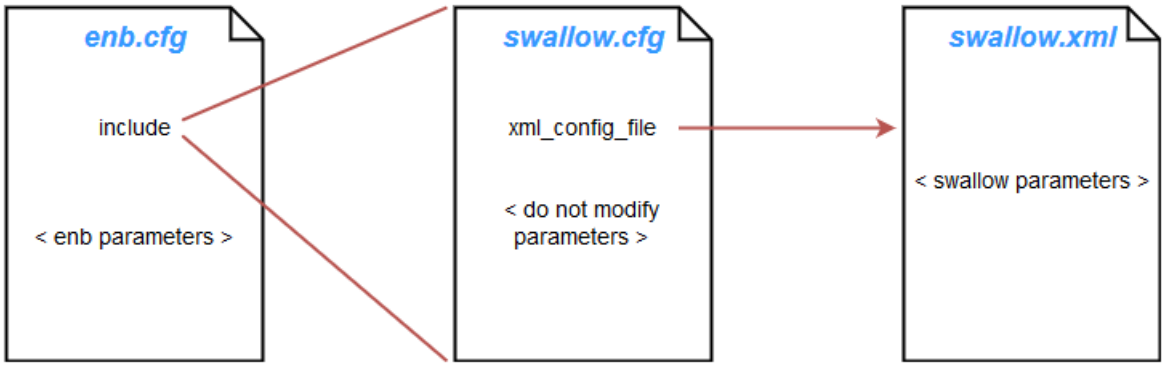


Figure 19: Configuration files linkage

11.1 Amarisoft LTEENB/LTEUE configuration file (enb.cfg or ue.cfg)

The Amarisoft LTEENB/LTEUE configuration file (`enb.cfg`, `gnb.cfg` or `ue.cfg`) is provided by Amarisoft and is used to configure the LTEENB/LTEUE software. The user should follow Amarisoft documentation for instructions on how to setup this file.
In this file, only the `include` directive needs to be modified to include the `swallow.cfg` driver interface file, as shown in Figure 20. This allows operation of the LTEENB/LTEUE software with the Swallow board as a RF driver.

```
/* RF driver configuration */
include "swallow.cfg",
```

Figure 20: Driver interface file selection

Note that if the Amarisoft LTEENB software has been installed with Amarisoft's SDR card support, it might be needed to change the `rf_driver` symbolic link inside the LTEENB `config` folder. Else, the LTE service might not launch the LTEENB application due to failure to detect the Amarisoft's SDR card. To change the link, simply execute the following command inside the LTEENB `config` folder:

```
./rf_select.sh swallow.cfg
```

11.2 Swallow driver interface file (swallow.cfg)

The Swallow driver interface file (`swallow.cfg`) is used to inform LTEENB/LTEUE software which RF driver to use. It includes some do-not-modify parameters for correct operation, as well as a link to the actual Swallow configuration file. An example of this file is available in the Swallow Installer package. Its default content is shown in Figure 21.

```
/*      Swallow V6
        Full transceiver configuration is done in the file pointed to by xml_config_file parameter
*/

rf_driver: {
        name: "swallow",
        //name: "swallow_mrat",
        path: "/usr/local/lib/lteenb",
        xml_config_file: "/root/enb/config/swallow.xml", /* Full path to XML configuration file */
},

/* DO NOT MODIFY the following parameters */
tx_gain_offset: -15.0,
tx_gain: 0.0,
rx_gain: 0.0,
tx_pad_duration: 0,
rx_ta_offset: 0.0,
tx_time_offset: 0,
```

Figure 21: Swallow driver interface file content

This file should be put into the same location (directory) as the LTEENB/LTEUE configuration file.

In this file, the `name` parameter can be changed from `swallow` to `swallow_mrat` depending on if the RF driver should be the legacy transceiver (`swallow`) or the Multi-RAT transceiver (`swallow_mrat`).
The usage of the Multi-RAT transceiver requires the RMU service to be running as the Radio Equipment C&M channels are not implemented inside the transceiver but inside the RMU.
In the other hand, if using the legacy transceiver, then the RMU service may not be running as the Radio Equipment C&M channels are implemented inside the transceiver, but in that case, the system can only operate a single RAT (here, the LTEENB).
This offers two modes of operation:

- Swallow legacy transceiver (`swallow`) + RMU deactivated (single-RAT operation only).
- Swallow Multi-RAT transceiver (`swallow_mrat`) + RMU activated (multi-RAT operation possible).

The parameter `xml_config_file` can be modified to set the full path to the Swallow XML configuration file.

Important: Make sure that the do not modify parameters defined in `swallow.cfg` are not overridden in `enb.cfg`. Changing these parameters values may reduce performance or, in the worst cases, cause irreversible damage to the connected Radio Equipment.

11.3 Swallow configuration file (swallow.xml)

The swallow configuration file (`swallow.xml`) is a XML file that includes all parameters than can be edited to configure the fronthaul portion of the RAN. An example of this file is available in the Swallow Installer package. Its default content is shown in Figure 22.

This file should be put at the location pointed to by the `xml_config_file` parameter of the Swallow driver interface file, and edited as per system requirements. The following sections provide details on how this file can be modified, with description of the different parameters, their acceptable values and admissible ranges.

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Swallow V6 LTEENB TRx PCIe configuration file -->

<swallow minor="0">

        <!-- CPU management -->
        <cpu wait-mode="poll" irq-interval-us="250"/>

        <!-- Synchronization scheme (ignored when using MRAT transceiver) -->
        <sync mode="gps" output="umts"/>

        <!-- CPRI master ports setup (ignored when using MRAT transceiver) -->
        <port id="0" cpri-line-speed="3"/>
        <port id="1" cpri-line-speed="3"/>
        <port id="2" cpri-line-speed="3"/>
        <port id="3" cpri-line-speed="3"/>

        <!-- UE mode: cells definition (only applicable when running 'lteue' application) -->
        <!--
        <ue_cell mode="fdd" n_rb_dl="100" dl_earfcn="1575" ul_earfcn="19575" n_antenna_dl="2" n_antenna_ul="2" duplicate_tx="false"/>
        <ue_cell mode="fdd" n_rb_dl="50" dl_earfcn="3100" ul_earfcn="21100" n_antenna_dl="2" n_antenna_ul="2" duplicate_tx="false"/>
        -->

        <!-- Cells configuration -->
        <cell id="0">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="30.0"/>
                <rx id="0" master-port="0" hop-count="0" antport="0"/>
                <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="30.0"/>
                <rx id="1" master-port="0" hop-count="0" antport="1"/>
        </cell>
        <cell id="1">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="1" hop-count="0" antport="0" power-dBm="30.0"/>
                <rx id="0" master-port="1" hop-count="0" antport="0"/>
                <tx id="1" master-port="1" hop-count="0" antport="1" power-dBm="30.0"/>
                <rx id="1" master-port="1" hop-count="0" antport="1"/>
        </cell>
        <cell id="2">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="2" hop-count="0" antport="0" power-dBm="30.0"/>
                <rx id="0" master-port="2" hop-count="0" antport="0"/>
                <tx id="1" master-port="2" hop-count="0" antport="1" power-dBm="30.0"/>
                <rx id="1" master-port="2" hop-count="0" antport="1"/>
        </cell>
</swallow>
```

Figure 22: Swallow configuration file content

Note that when using the Swallow Multi-RAT transceiver (`swallow_mrat`), the Swallow `minor` argument can be set to select which Swallow board should be used (possibly each cell being handled by a different board, see Annex C: Example multi-board LTEENB configuration) by the RF driver, in the case where multiple Swallow boards are plugged in the host computer. The minor argument is ignored when using the Swallow legacy transceiver (`swallow`) and the first board available (`minor="0"`) is always used.

11.3.1 CPU management

Element:
`swallow -> cpu`

Cardinality:
Unique.

Description:
Due to real-time requirements of the LTEENB/LTEUE software and the high CPU usage that comes from running a fully software-running LTE eNodeB, there may be needs to tune the Swallow transceiver to lessen CPU usage as much as possible. This element provides parameters to configure such optimizations. These parameters are optional.

Parameter	Type	Description
`wait-mode`	Enumerated string	Sets the CPU wait mode as either `poll` or `irq`. Polling mode typically offers best performance on real-time kernels. Default: `poll`
`irq-interval-us`	Unsigned integer	Applicable only when `wait-mode` is set to `irq`. When using higher sample rates (wider channel bandwidths), the number of interrupt requests issued by the PCIe board increases. Because interruptions cause CPU context switches, CPU usage increases. This parameter defines the minimum interval time, in microseconds, between each interrupt request issued by the PCIe. Thus, increasing values for this parameter may reduce CPU usage. For a value of 500, the board will issue an interruption every 500µs or longer. For a value of 0, the actual interrupt request interval is proportional to the channel bandwidth (around 133µs in LTE 20MHz, 266µs in LTE 10MHz). Note that higher values also increase latency by the given amount. Range: 0 to 2000 [µs] Default: 250 [µs]

Table 8: CPU management parameters

11.3.2 Synchronization scheme

Element:
`swallow -> sync`

Cardinality:
Unique.
Element is ignored when using the Multi-RAT transceiver.

Description:
This element allows configuration of the time and frequency synchronization schemes used and output by the PCIe board. An external time-synchronization source is highly recommended to operate and prevent interferences with adjacent radio nodes (especially in TDD transmission mode).

Parameter	Type	Description
`mode`	Enumerated string	Selects the synchronization source used by the board from the following values: `freerun`: The board runs on its internal clock oscillator and is not synchronized to any external source. `refclk`: The board synchronizes itself on an external 15.36 MHz reference clock from the SYNC_IN connector, providing frequency synchronization only. Invalid for type D, E and F boards. `gps`: The board synchronizes itself on the GPS system. Requires the usage of a GPS antenna on the SYNC_IN connector, providing time and frequency synchronization. `umts`: The board synchronizes itself on an external 100Hz UMTS synchronization pulse (e.g. from another Swallow board) on the SYNC_IN connector, providing time and frequency synchronization. Invalid for type D, E and F boards. `pps-rise`: The board synchronizes itself on the rising edge of an external PPS synchronization pulse on the SYNC_IN connector, providing time and frequency synchronization. Invalid for type D, E and F boards. `pps-fall`: The board synchronizes itself on the falling edge of an external PPS synchronization pulse on the SYNC_IN connector, providing time and frequency synchronization. Invalid for type D, E and F boards.

`output`	Enumerated string	Selects the synchronization signal type output from the board (not applicable to type D, E and F boards) that can be used to synchronize other nodes, from the following values: `reflck`: A 15.36 MHz clock is output on the J8 connector, providing frequency synchronization only to another board. `umts`: A 100Hz UMTS synchronization pulse is output on the J8 connector, providing time and frequency synchronization to another board.

Table 9: Synchronization scheme parameters

11.3.3 CPRI master ports

Element:
`swallow -> port`

Cardinality:
One for each CPRI port populated by firmware.
Elements are ignored when using the Multi-RAT transceiver.

Description:
This elements provides CPRI port specific configuration.

Parameter	Type	Description
`id`	Unsigned integer	Selects which CPRI port this element refers to (indexed from zero). For instance, a value of 2 means that this element refers to the board's third CPRI port (i.e. CPRI interface on SFP_2 connector). Range: 0 to NUM_CPRI_PORTS-1

`cpri-line-speed`	Unsigned integer	Sets the CPRI line bit rate option to be used on the selected Swallow CPRI port. This line bit rate option is also configured on all Radio Equipment CPRI master ports connected to this Swallow port (relevant only in daisy-chaining topology cases). Range (type A board): 2 to 6 Range (type B, C, D, E and F boards): 2 to 8 *Note 1: Range may be further restricted by the line bit rate options supported by Radio Equipment and used SFP transceiver.* *Note 2: Due to different encoding, it is not possible to mix line bit rate option 8 with other line bit rates on the same board.*

Table 10: CPRI master ports parameters

11.3.4 UE mode cells definition

Element:
`swallow -> ue_cell`

Cardinality:
One for each UE cell configured in LTEUE software.
These elements are only applicable when running LTEUE application.

Description:
This element allows definition of UE radio cells. Because the Amarisoft TRX API cannot provide full UE cell definition, this element must be full configured for correct Radio Equipment parameterizing.

Parameter	Type	Description
`mode`	Enumerated string	Duplex mode configuration. Must be either `fdd` or `tdd`.
`n_rb_dl`	Unsigned integer	Number of resource blocks in downlink (Rx) direction. Defines the channel bandwidth, can be either: `1`: 0.2 MHz `6`: 1.4 MHz `15`: 3 MHz `25`: 5 MHz `50`: 10 MHz `75`: 15 MHz `100`: 20 MHz
`n_rb_ul`	Unsigned integer	Same as `n_rb_dl` but in uplink (Tx) direction.
`dl_earfcn`	Unsigned integer	Downlink (Rx) EARFCN.
`ul_earfcn`	Unsigned integer	Uplink (Tx) EARFCN.
`n_antenna_dl`	Unsigned integer	Number of downlink (Rx) antennas for the UE cell.
`n_antenna_ul`	Unsigned integer	Number of uplink (Tx) antennas for the UE cell.
`duplicate_tx`	Boolean	Can be either `true` or `false`. In UE mode of operation, uplink (Tx) data is usually transmitted only on a single antenna. This parameter allows, when set to `true`, the copy of I/Q samples transmitted on uplink primary antenna to the uplink secondary antennas. This may improve uplink total power and increase range/throughput.

Table 11: UE mode cells defintion

11.3.5 Cells

Element:
`swallow -> cell`

Cardinality:
One for each cell configured in LTEENB/LTEUE software.

Description:
This element is the parent element for a specific radio cell's parameterizing. Actual configuration for the relevant cell is deferred to the child elements, described below.

Parameter	Type	Description
`id`	Unsigned integer	Selects which configured cell this element refers to (indexed from zero). For instance, a value of 0 means that this element refers to the first cell configured in LTEENB software. Range: 0 to NUM _CELLS-1 *Note: Swallow cell ids are incremented in the order the cells are declared in the LTEENB software configuration file. It is unrelated to the actual `cell_id` parameter of that file.*

Table 12: Cells parameters

11.3.5.1 Cell I/Q compression

Element:
`swallow -> cell -> iq-compression`

Cardinality:
Unique per cell

Description:
This element allows parameterizing of the radio cell I/Q data compression. I/Q data compression is an optional digital signal processing method that can be used to reduce PCIe and CPRI bandwidth usage, as well as CPU usage, by reducing the sample rate and applying a non-linear quantization process to reduce the number of bits needed to represent an I/Q sample.

Parameter	Type	Description
`type`	Enumerated string	Selects the I/Q compression type used for all signal paths (Tx and Rx) of this radio cell. The following values can be used: `none`: No I/Q compression of any form is used. `lossless`: Only sample rate is reduced, no performance degradation is expected. `full`: Sample rate is reduced and the non-linear quantization process is applied. The non-linear quantization typically induces a performance degradation. *Note 1: The usage of I/Q compression may be restricted by the capabilities of the Radio Equipment.* *Note 2: While I/Q compression is restricted to LTE 10MHz, LTE 15MHz and LTE 20MHz channel bandwidths, sample rate reduction only applies to LTE 10MHz and LTE 20MHz channel bandwidths.* *Note 3: Sample rate reduction reduces CPU usage, PCIe bandwidth and CPRI bandwidth by 25%. Non-linear quantization further reduces CPRI bandwidth usage by 33%.* *Note 4: When available, it is always recommended to use lossless I/Q compression, as it provides great improvements without drawbacks.*
`tx-sigma`	Unsigned integer	Non-linear quantization "σ" parameter for Tx signal paths of this radio cell, relevant only when using full I/Q compression. Range: 0 to 65535 *Note: Due to known signal power dynamics in the Tx direction, the default value of 7000 is optimal and there is typically no need to modify it. No performance degradation is to be expected.*

`rx-sigma`	Unsigned integer	Non-linear quantization "σ" parameter for Rx signal paths of this radio cell, relevant only when using full I/Q compression. Range: 0 to 65535 *Note: Due to the wide range of signal powers that may be acceptable in the Rx direction, this parameter may influence performance. Typically, lower values will favor low amplitude signals (far signal source), while higher values will favor high amplitude signals (close signal source) at the cost of reduced sensitivity.*

Table 13: Cell I/Q compression parameters

11.3.5.2 Cell time offset

Element:
`swallow -> cell -> time-offset`

Cardinality:
Unique per cell, optional

Description:
This optional element offers the possibility to time-shift a radio cell's I/Q data relative to the board's time reference (i.e. GPS time).

Parameter	Type	Description
`value-us`	Float	Time shift value (in microseconds) relative to the absolute time reference for this cell. A negative value means that radio subframe 0 will be transmitted before the global start of 10ms radio frame marker. A positive value means that radio subframe 0 will be transmitted after the global start of 10ms radio frame marker. Range: -10000 to 10000 [µs] Default: 0 [µs] *Note: Due to the limited processing time between Rx and Tx samples imposed by the LTEENB software (around 3 milliseconds), care should be taken when selecting the time offset value in multi-sector and/or multi-cell eNodeB. The*

		LTEENB software processing time will be reduced by the maximum time-offset difference between cells. *For instance, if one cell is set to a time offset of +500µs, and another cell is set to a time offset of -1100µs, then the LTEENB processing time is reduced by a total of 1600µs. Too large differences can cause Tx Underflow and/or Rx Overflow conditions to appear.*

Table 14: Cell time offset parameters

11.3.5.3 Cell Tx signal paths

Element:
`swallow -> cell -> tx`

Cardinality:
One for each Tx antenna configured for the parent cell in LTEENB/LTEUE software.

Description:
This element allows configuration of a specific Tx signal path for the parent cell.

Parameter	Type	Description
`id`	Unsigned integer	Selects which Tx antenna path for the parent cell this element refers to (indexed from zero). For instance, a value of 1 means that this element refers to the second Tx antenna path for the parent cell configured in LTEENB/LTEUE software. Range: 0 to NUM_TX_ANTS_ON_PARENT_CELL-1
`master-port`	Unsigned integer	Selects on which Swallow CPRI master port this Tx signal path is routed to (indexed from zero). Range: 0 to NUM_CPRI_PORTS-1
`hop-count`	Unsigned integer	Selects the number of hops (in CPRI daisy-chaining topology) this Tx signal path must go through from the REC before being terminated by the target RE for emission. Range: 0 to 255
`antport`	Unsigned integer	Selects on which RE physical antenna port this Tx signal path is mapped to (indexed from zero). Range: 0 to NUM_RE_ANT_PORTS-1

power-dBm	Float	Configures the maximum output power (in dBm) for this Tx signal path. Range: dependent on RE capabilities.

Table 15: Cell Tx (downlink) signal paths parameters

11.3.5.4 Cell Rx signal paths

Element:
`swallow -> cell -> rx`

Cardinality:
One for each Rx antenna configured for the parent cell in LTEENB/LTEUE software.

Description:
This element allows configuration of a specific Rx signal path for the parent cell.

Parameter	Type	Description
id	Unsigned integer	Selects which Rx antenna path for the parent cell this element refers to (indexed from zero). For instance, a value of 1 means that this element refers to the second Rx antenna path for the parent cell configured in LTEENB/LTEUE software. Range: 0 to NUM_RX_ANTS_ON_PARENT_CELL-1
master-port	Unsigned integer	Selects on which Swallow CPRI master port this Rx signal path is routed from (indexed from zero). Range: 0 to NUM_CPRI_PORTS-1
hop-count	Unsigned integer	Selects the number of hops (in CPRI daisy-chaining topology) this Rx signal path must go through from the target RE before being terminated by the REC for reception. Range: 0 to 255
antport	Unsigned integer	Selects on which RE physical antenna port this Rx signal path is mapped from (indexed from zero). Range: 0 to NUM_RE_ANT_PORTS-1

Table 16: Cell Rx (uplink) signal paths parameters

11.3.6 Custom channel filtering

Custom FIR filtering in the Swallow V6 transceiver is supported only on a specific firmware (image loaded into the board) and software (driver) version. Namely, the firmware minor version needs to be >= 8, and must contain the "F" suffix (for "F"iltering) in the firmware ordering information, i.e.:
`swallow_V6B8_2p_2c22F.bit` or `swallow_V6C8_3p_3c422F.bit`

This feature is experimental and is typically not required for standard deployments, as the channel filtering is usually done by the Radio Equipment instead of the eNodeB.

Each cell antenna channel can support its own, independent, upto 512-taps complex FIR filter. This is done by setting the "`filter`" attribute in the `swallow->cell->tx` or `swallow->cell->rx` element to indicate the full path of the file containing the filter's impulse response coefficients.

For instance, Figure 23 shows how to setup the Swallow XML configuration file so that the TX0 path uses filter "*filter1.txt*" and RX0 path uses "*filter2.txt*" (also note that TX1/RX1 are not configured to use any filter):

```
<cell id="0">
        <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
        <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="30.0" filter="/home/user/filter1.txt"/>
        <rx id="0" master-port="0" hop-count="0" antport="0" filter="/home/user/filter2.txt"/>
        <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="30.0"/>
        <rx id="1" master-port="0" hop-count="0" antport="1"/>
</cell>
```

Figure 23: Selecting FIR filters

In this example, the files `/home/user/filter1.txt` and `/home/user/filter2.txt` contain the filters' impulse response coefficients in text format. The file format is explained in section Filter impulse response file format of this document.

When the LTEENB is started, the console should show the FIR filters being applied to the transceiver signal processing, as shown in Figure 24:

```
Swallow Multi-RAT transceiver version 0.12
RF0: sample_rate=15.360 MHz dl_freq=881.500 MHz ul_freq=836.500 MHz (band 5) dl_ant=2 ul_ant=2
(enb) [CELL5] Calibrating PCIe bandwidth for Swallow board 0... done
        Tx burst order: 5, available bandwidth: 7641 Mb/s
        Rx burst order: 0, available bandwidth: 7999 Mb/s
[CELL0_TX0] 512-taps filter configured
[CELL0_RX0] 128-taps filter configured
```

Figure 24: LTEENB console with FIR filters being configured

An important note to make when using custom FIR filters is that any filtering will cause some kind of group delay to the signal. This latency needs to be compensated inside the baseband unit, else the timing advance computation will be incorrect (the user equipment will be estimated "farther" than they really are).
However, as this group delay is a direct function of the coefficients loaded in the filter, the group delay value needs to be entered manually. This can be done by setting the "`delay-comp-us`" attribute in the `swallow->cell->tx` or `swallow->cell->rx` elements. The user should then input the correct group delay (in microseconds) into these attributes. Figure 25 shows the syntax for this.

```
<cell id="0">
        <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
        <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="30.0" filter="/home/user/filter1.txt" delay-comp-us="2.0"/>
        <rx id="0" master-port="0" hop-count="0" antport="0" filter="/home/user/filter2.txt" delay-comp-us="1.234"/>
        <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="30.0"/>
        <rx id="1" master-port="0" hop-count="0" antport="1"/>
</cell>
```

Figure 25: Setting delay compensation

The complete LTEENB console startup output for this setup is displayed in Figure 26.

```
Swallow Multi-RAT transceiver version 6.12
RF0: sample_rate=15.360 MHz dl_freq=881.500 MHz ul_freq=836.500 MHz (band 5) dl_ant=2 ul_ant=2
(emb) [CELL5] Calibrating PCIe bandwidth for Swallow board 0... done
        Tx burst order: 5, available bandwidth: 7628 Mb/s
        Rx burst order: 2, available bandwidth: 8660 Mb/s
[CELL0_TX0] 512-taps filter configured
[CELL0_TX0] user delay compensation: 2.000µs
[CELL0_RX0] 128-taps filter configured
[CELL0_RX0] user delay compensation: 1.234µs
```

Figure 26: LTEENB console with FIR and delay compensation

11.3.6.1 Filter impulse response file format

The filter's impulse response coefficient file is a text file that supports two formats:
1. Real-valued coefficients, or
2. Complex-valued coefficients.

It is not possible to mix both formats in the same file.

For either format, each line represents a coefficient (or tap), and a maximum of 512 coefficients is supported by software and firmware. There can be either odd or even number of coefficients and symmetry is not required.

In real format, each line contains a single decimal value, representing the real-valued coefficient. For instance, the content of the file shown in Figure 27 is a valid symmetric, real, 21-taps filter.

real-filter.txt

```
1  -0.02010411882885732
2  -0.05842798004352509
3  -0.061178403647821976
4  -0.010939393385338943
5  0.05125096443534972
6  0.033220867678947885
7  -0.05655276971833928
8  -0.08565500737264514
9  0.0633795996605449
10 0.31085440365663597
11 0.4344309124179415
12 0.31085440365663597
13 0.0633795996605449
14 -0.08565500737264514
15 -0.05655276971833928
16 0.033220867678947885
17 0.05125096443534972
18 -0.010939393385338943
19 -0.061178403647821976
20 -0.05842798004352509
21 -0.02010411882885732
```

Figure 27: Example real-valued file format

In complex format, each line contains two decimal values separated by a comma. The left decimal value is the real part, and the right decimal value is the imaginary part of the complex coefficient (that is: `<real>,<imag>`). For instance, the content of the file shown in Figure 28 is a valid asymmetric, complex, 32-taps filter.

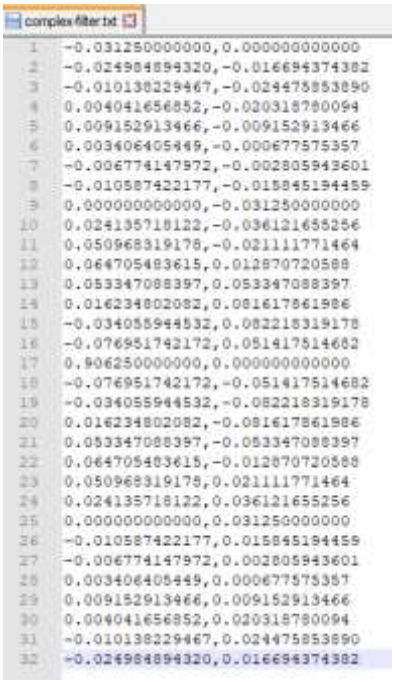


Figure 28: Example complex-valued file format

There should be no trailing whitespaces or empty lines in the file (even at the end) else the transceiver will fail to load the filter. If file parsing fails, the LTEENB console will display a corresponding error message and abort startup.

Moreover, care must be taken when designing the filter. The resulting filter should not have important gain in the pass-band, else signal saturation may occur. Even though Radio Equipments typically implement overdrive protection mechanisms, some edge cases with saturated signals may damage the power amplifiers. It is recommended to make sure that the overall pass-band gain is not designed for a gain above unity gain (0 dB).

12 User interface (SNMP/CLI)

The system user interface is provided by the Radio Management Unit, where all management parameters reside.

The RMU offers two modes of interfacing:

- Command Line Interface (CLI): which are commands entered directly on the RMU console. This mode is typically used for debugging or testing purposes.
- Simple Network Management Protocol (SNMP): where the RMU extends the SNMP agent daemon as a SNMP AgentX subagent. This mode is preferred in a production environment as it allows the system to be managed remotely from a NMS.

Both modes are equivalent in terms of management capabilities, and both can be safely used concurrently.

Important: The RMU SNMP AgentX subagent implementation does not support SNMP SET commands with multiple variable bindings. SNMP SET commands must be used with a single variable only.

This section is provided as the reference documentation for the RMU system's managed objects, it describes all objects defined in the RMU SNMP Management Information Base (MIB). The MIB file `AW2S-RMUv6-MIB.txt` is available in the `snmp` directory of the Swallow V6 installer package. For most of those SNMP-managed parameters, an equivalent CLI command exists on the RMU console.

In below documentation, color codes are used to help the reader identify the type of the different MIB objects. Table 17 lists and describes those color codes.

Textual convention	Type definition with more precise semantics compared to primitive types defined in the SMI. Used for the convenience of humans reading the MIB module.
Read-Only Variable	Defines a variable for which only GET are possible. The variable cannot be SET.
Read-Write Variable	Defines a variable for which both GET and SET are possible.
Read-Write Command	Defines a variable for which both GET and SET are possible. Writes (SETs) are interpreted as "commands" which have special meanings and side effects.
Notification	Asynchronous (agent-generated) message sent to the supervisor to inform an event occurred. Also named "trap" in SNMP v1.
Notification Payload	Payload data associated to a notification message as a variable binding.

Table 17: Variables color coding

Moreover, for the purpose of redacting this document, extracts from SNMP traps are realized by running the `snmptrapd` daemon locally on the host, via following command:

```
sudo snmptrapd -f -Lo -m ALL
```

The content of the `snmptrapd.conf` file is set to:

```
authCommunity log,execute,net public
format2 %V\n %v\n\n
```

12.1 Textual conventions

The following textual conventions are relevant for multiple elements of the user interface, where they precisely define common data types, their syntax and usage.

Textual convention	RMUObjectStatus
Syntax	INTEGER (enumerated)
Description	A value that represents the status of a RMU object. ➢ none(0) : Status information does not apply to the object or is not yet available. ➢ disabled(1) : Object is voluntarily disabled based on user configuration or parameters. When in this state, the object is not operating. ➢ errored(2) : Object is in error state, either due to wrong configuration, or due to external events. This state typically indicates a fault that prevents correct operation for the object. ➢ preOperational(3) : Object is preparing for operation, indicating that the object is functioning normally but requires some time or external event in order to enter the fully operational state. This status can also indicate that the object is initializing. ➢ operational(4) : Object is fully operational, in this state the object is fully configured, initialized, and operates as expected with regards to its parameters and external events. The object status is always a read-only value and an object may or may not support all of these enumerated values. RMUObjectStatus values are typically accompanied with a status information string that provides extra information regarding the current object status. The RMUObjectStatus value is usually updated asynchronously.

Textual convention	OriFST
Syntax	INTEGER (enumerated)
Description	A value that represents the functional state of an object residing inside a radio equipment, as specified in the Open Radio Interface. ➢ preOperational(0) : Object is preparing for operation but not yet able to provide the expected service. (e.g. warming up, calibrating, etc.). ➢ operational(1) : Object is fully functional. ➢ degraded(2) : Object operates, but deviates from expected performance. ➢ failed(3) : Object is faulty and not operating. ➢ notOperational(4) : Object is available, but not in operation. ➢ disabled(5) : Object is unavailable for operation, e.g. because of not equipped hardware resources. ➢ unknown(6) : Object functional state is unknown.

Textual convention	InfoString
Syntax	OCTET STRING
Description	Same as DisplayString defined in SNMPv2-TC, but not limited in length.

Textual convention	DirNameString
Syntax	OCTET STRING (SIZE(1..255))
Description	A non-empty character string representing a file system directory name. The naming convention for directory names is specified in order to increase security and interoperability. For a directory name, any of the following characters are allowed: ❖ Unreserved characters set as defined in RFC3986, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~. ❖ The character / that acts as directory arborescence delimiter. ❖ Any other character is forbidden. The character string is validated by the following regular expression: [^a-zA-Z0-9-._~/]

Textual convention	FileNameString
Syntax	OCTET STRING (SIZE(1..63))
Description	A non-empty character string representing a file system file name. The naming convention for file names is specified in order to increase security and interoperability. A file name string must respect the following: ❖ The first character is an alphanumeric character in the ranges a-z, A-Z and 0-9. ❖ Other characters can be any character from the unreserved characters set as defined in RFC3986, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~. ❖ Any other character is forbidden. The character string is validated by the following regular expression: [^a-zA-Z0-9-._~]

Textual convention	FileNameOrEmptyString
Syntax	OCTET STRING (SIZE(0..63))
Description	A potentially empty character string representing a file system file name. The naming convention for file names is specified in order to increase security and interoperability. A non-empty file name string must respect the following: ❖ The first character is an alphanumeric character in the ranges a-z, A-Z and 0-9. ❖ Other characters can be any character from the unreserved characters set as defined in RFC3986, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~. ❖ Any other character is forbidden.

	An empty file name string may have different meanings depending on context. The character string is validated by the following regular expression: `[^a-zA-Z0-9-._~]`

Textual convention	**FilePathString**
Syntax	`OCTET STRING (SIZE(1..255))`
Description	A non-empty character string representing a file system absolute file path. A file path includes both the file's directory location in the file system and the file name, separated by a / delimiter. The naming convention for file paths is specified in order to increase security and interoperability. For a file path, any of the following characters are allowed: ❖ Unreserved characters set as defined in RFC3986, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~. ❖ The character / that acts as directory arborescence delimiter. ❖ Any other character is forbidden. The character string is validated by the following regular expression: `[^a-zA-Z0-9-._~/]`

Textual convention	**AliasString**
Syntax	`OCTET STRING (SIZE(1..63))`
Description	A non-empty character string describing either the name or the alias of an object's instance. An alias string is usually arbitrary and only serves as a quick reference for the network management. The alias string context's reside in the application and does not relate to any file system naming convention. An alias string must respect the following: ❖ Characters can be any character from the unreserved or reserved characters set as defined in RFC3986, excepted XML predefined entities, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~:/?#[]@!$()*+,;=. ❖ Any other character is forbidden. The character string is validated by the following regular expression: `[^a-zA-Z0-9-._~:/?#[\]@!$()*+,;=]`

Textual convention	**SingleFileTransferString**
Syntax	`OCTET STRING`
Description	A character string composed of an `AliasString` followed by a whitespace then a `FileNameString`. The `AliasString` corresponds to a valid FTP link instance either represented by its object name (in the form `ftp/link:X`) or its user-specified alias.

Textual convention	**CredentialString**
Syntax	`OCTET STRING (SIZE(1..63))`
Description	A non-empty character string describing a credential such as a login user name or a password. The naming convention for credentials is specified in order to increase security and interoperability. A credential string must respect the following: ❖ Characters can be any character from the unreserved or reserved characters set as defined in RFC3986, excepted XML predefined entities, and excepted the shell command delimiter ;, i.e. alphanumeric characters in the ranges a-z, A-Z and 0-9 and graphic symbols -._~:/?#[]@!$()*+,=. ❖ Any other character is forbidden. The character string is validated by the following regular expression: `[^a-zA-Z0-9-._~:/?#[\]@!$()*+,=]`

12.2 System level monitoring

The system level monitoring provides simple read-only elements that inform of the RMU application status.
It displays application startup arguments, uptime, and most importantly, provides a Master Alarm feature that proves useful for the detection of any error status occurring in the RMU.

Read-Only Variable	rmuArgs
SNMP OID	1.3.6.1.4.1.50488.2.1.0
Syntax	DisplayString
Description	Radio Management Unit application command line arguments.

Read-Only Variable	rmuVersion
SNMP OID	1.3.6.1.4.1.50488.2.2.0
Syntax	DisplayString
Description	Radio Management Unit software version.
CLI equiv.	info

Read-Only Variable	rmuUptime
SNMP OID	1.3.6.1.4.1.50488.2.3.0
Syntax	TimeTicks
Description	The time (in hundredths of a second) since the Radio Management Unit was last started.
CLI equiv.	info

Notification	rmuStarting
SNMP OID	1.3.6.1.4.1.50488.2.5
Description	The RMU starting notification is sent by the RMU application at startup.

Notification	rmuStopping
SNMP OID	1.3.6.1.4.1.50488.2.6
Variables	rmuUptime
Description	The RMU stopping notification is sent by the RMU application when exiting, either due to an error or upon user request. The rmuUptime variable binding provides information about the duration the RMU application was running before stopping.

The RMU master alarm, defined below, multiplexes the statuses of all objects in the RMU (configuration manager, RAT C&M channels and services, Swallow boards, Radio Equipments…) into a single variable.
The NMS may choose to monitor the rmuMasterAlarm variable or react upon rmuMasterAlarmSet and rmuMasterAlarmCleared notifications to identify if an issue is currently active. Note that this alarm does not provide information about the type of error that occurred; it merely serves as an indication that an error is active somewhere.

For instance, Figure 29 shows the RMU console master alarm being set, following the loss of an Ethernet over CPRI interface:

Figure 29: Console logs showing RMU master alarm being set

Resulting in the following SNMP trap being captured by the NMS:

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (36776) 0:06:07.76
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortDhcpStatusChanged
 AW2S-RMUv6-MIB::swallowPortDhcpStatus.0.2 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::swallowPortDhcpStatusInfo.0.2 = STRING: "dhcpdump: pcap_loop(swa0p2): The interface went down"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (36794) 0:06:07.94
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::rmuMasterAlarmSet
 AW2S-RMUv6-MIB::rmuMasterAlarm.0 = INTEGER: true(1)
```

Figure 30: RMU master alarm trap

Read-Only Variable	**rmuMasterAlarm**
SNMP OID	1.3.6.1.4.1.50488.2.4.0
Syntax	TruthValue
Description	Indicates whether the RMU master alarm is set or not. The value true(1) means that at least one of the RMU objects' RMUObjectStatus value is errored(2). Thus, the master alarm provides a quick and easy way to figure out whether an error occurred at any level.
CLI equiv.	info

Notification	**rmuMasterAlarmSet**
SNMP OID	1.3.6.1.4.1.50488.2.7
Variables	rmuMasterAlarm
Description	A rmuMasterAlarmSet notification signifies that at least one of the RMU objects' RMUObjectStatus has just been changed to errored(2). Monitoring this notification is the easiest way to detect errors occurring in the RMU.

Notification	**rmuMasterAlarmCleared**
SNMP OID	1.3.6.1.4.1.50488.2.8
Variables	rmuMasterAlarm
Description	A rmuMasterAlarmCleared notification signifies that the RMU exited the state where the master alarm is set. That is to say when all RMU objects' RMUObjectStatus are not errored(2) anymore.

12.3 Logging and display parameters

The logging and display parameters influence how the RMU writes logs into the log file and on the console.

Read-Only Variable	logStatus
SNMP OID	1.3.6.1.4.1.50488.2.10.1.0
Syntax	RMUObjectStatus
Description	RMU log status.
CLI equiv.	log info

Read-Only Variable	logStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.10.2.0
Syntax	InfoString
Description	RMU log status information details. May be empty if no relevant information is available.
CLI equiv.	log info

Notification	logStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.10.10
Variables	logStatus, logStatusInfo
Description	This notification is emitted when the RMU log status has changed. The included values of logStatus and logStatusInfo provide the new status and possibly textual information regarding this new status.

Among the following parameters, logDirName (which is set by the RMU application startup argument) and logFileName (which can be changed at any time), define the location of the log file logFilePath.

```
user@shuttle:~$ snmpget -v 2c -c public localhost -m ALL logDirName.0 logFileName.0 logFilePath.0
AW2S-RMUv6-MIB::logDirName.0 = STRING: "/var/log/rmu"
AW2S-RMUv6-MIB::logFileName.0 = ""
AW2S-RMUv6-MIB::logFilePath.0 = STRING: "/var/log/rmu/2021-03-26_10-11-09__0.log"
user@shuttle:~$ ls /var/log/rmu/
2021-03-26_10-11-09__0.log
user@shuttle:~$
```

Figure 31: Retrieving log file name

When the log file changes (either due to a write to logFileName or when the log file wraps), the logFileChanged notification is emitted by the RMU.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost logFileName.0 s "myFile.log"
AW2S-RMUv6-MIB::logFileName.0 = STRING: "myFile.log"
user@shuttle:~$ ls /var/log/rmu/
2021-03-26_10-11-09__0.log  myFile.log
user@shuttle:~$
```

Figure 32: Setting a new log file name

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (60857) 0:10:08.57
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::logFileChanged
 AW2S-RMUv6-MIB::logFilePath.0 = STRING: "/var/log/rmu/myFile.log"
```

Figure 33: Log file changed notification

Read-Only Variable	logDirName
SNMP OID	1.3.6.1.4.1.50488.2.10.3.0
Syntax	DirNameString
Description	RMU log working directory. This is the directory where new log files are written.
CLI equiv.	log info -v

Read-Write Variable	logFileName
SNMP OID	1.3.6.1.4.1.50488.2.10.4.0
Syntax	FileNameOrEmptyString
Description	Specifies RMU log file name. If the string is empty, then the RMU generates a file name based on RMU start date and time. RMU-generated file names also provide file wrapping capability where a new file is created (with some index number in the resulting file name) once the file size reaches the value set by logMaxFileSize. On file wrapping, the logFilePath is updated accordingly.
CLI equiv.	log file-name [<fileName>] log unset-file-name

Read-Only Variable	logFilePath
SNMP OID	1.3.6.1.4.1.50488.2.10.5.0
Syntax	FilePathString
Description	RMU log file absolute file path in the host's file system. May be updated when the log file wraps.
CLI equiv.	log info

Read-Write Variable	logMaxFileSize
SNMP OID	1.3.6.1.4.1.50488.2.10.6.0
Syntax	Gauge32 (Mbytes)
Description	RMU log file maximum size in Mbytes. A value of 0 means that the file size is unlimited. Relevant only when the log file can wrap, that is when logFileName is empty. The file size is unlimited regardless of this value when logFileName is not empty.
CLI equiv.	log max-file-size [<max-size_MB>]

Notification	logFileChanged
SNMP OID	1.3.6.1.4.1.50488.2.10.11
Variables	logFilePath
Description	This notification signifies that the current log file has changed. Usually following a write to the logFileName or when the log file wraps. The new log file path is included in the logFilePath value.

Finally, the following parameters are available to change how the logs are displayed in the RMU console and log file.

Read-Write Variable	logColorsEnable
SNMP OID	1.3.6.1.4.1.50488.2.10.7.0
Syntax	TruthValue
Description	RMU log colors enablement. When enabled, ANSI color escape sequences are used to make the log file data and console outputs more pretty and readable. ANSI escape sequences may or may not be supported by some terminals and log parsing tools.
CLI equiv.	log colors-enabled [true|false]

Read-Write Variable	logConsoleMute
SNMP OID	1.3.6.1.4.1.50488.2.10.8.0
Syntax	TruthValue
Description	RMU console muting. When muted, no text is output to the RMU console standard output.
CLI equiv.	log console-mute log console-resume

Read-Write Variable	logConsoleMinLevel
SNMP OID	1.3.6.1.4.1.50488.2.10.9.0
Syntax	INTEGER (enumerated)
Description	Minimum log severity for output to RMU console enablement. ➢ debug(0) : Debugging information, usually diagnostically helpful, aimed at resolving bugs or various issues and thoroughly understand the current state of the system. ➢ info(1) : Generally helpful information that should be logged under normal conditions. Information that provide top-level context for understanding current operation and warnings or errors that also occur. ➢ warn(2) : Meaningful information related to possible issues that usually does not require operator intervention but can prove useful at understanding the root cause of a subsequent error. ➢ error(3) : Important information that relates to runtime errors or unexpected conditions impacting the functionality of the system. Error conditions usually require operator intervention. The levels are sorted in increasing order of severity where debug(0) is the least impacting to the functionality of the RMU and error(3) is the most impacting: debug(0) < info(1) < warn(2) < error(3). When debug(0) is selected, all log entries are output to RMU console standard output.
CLI equiv.	log console-min-level [debug|info|warn|error]

12.4 Non-volatile RMU configuration

The RMU offers a non-volatile configuration file in XML format (detailed in section RMU configuration file (rmu.xml)). This configuration file defines default parameters for some elements managed in the RMU (such as CPRI port configuration).

However, it is possible that, for some reason, the configuration file could not be read successfully, or with some issues. In that case, the RMU configuration object status provide information about the error.

Read-Only Variable	cfgStatus
SNMP OID	1.3.6.1.4.1.50488.2.20.1.0
Syntax	RMUObjectStatus
Description	RMU configuration status.
CLI equiv.	cfg info

Read-Only Variable	cfgStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.20.2.0
Syntax	InfoString
Description	RMU configuration status information details. May be empty if no relevant information is available.
CLI equiv.	cfg info

Notification	cfgStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.20.7
Variables	cfgStatus, cfgStatusInfo
Description	This notification is emitted when the RMU configuration status has changed. The included values of cfgStatus and cfgStatusInfo provide the new status and possibly textual information regarding this new status.

Among the following parameters, cfgDirName (which is set by the RMU application startup argument) and cfgFileName (which can be changed by using the cfgSave command), define the location of the XML configuration file cfgFilePath.

More importantly, the configuration of some objects in the RMU can be changed during runtime by the Network Management using SNMP SETs or via CLI, and in some cases it may prove useful to store the updated configuration into non-volatile memory, so that in the future, the RMU can be restarted whilst keeping the new configuration.

Storing, or saving, the updated configuration can be done by using the cfgSave command, as shown in Figure 34 below. The first SET fails to due to the file naming convention not being respected, as displayed with a subsequent GET. The second SET is successful, and the current RMU configuration is saved into myConfig.xml file.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost cfgSave.0 s "An Invalid File Name"
Error in packet.
Reason: (badValue) The value given has the wrong type or length.
Failed object: AW2S-RMUv6-MIB::cfgSave.0

user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost cfgSave.0
AW2S-RMUv6-MIB::cfgSave.0 = STRING: "File name rules not respected"
user@shuttle:~$
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost cfgSave.0 s "myConfig.xml"
AW2S-RMUv6-MIB::cfgSave.0 = STRING: "myConfig.xml"
user@shuttle:~$
```

Figure 34: Saving the current RMU configuration

The `cfgFileChanged` notification is emitted by the RMU, indicating the configuration has been saved to a new file:

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (74742) 0:12:27.42
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cfgFileChanged
 AW2S-RMUv6-MIB::cfgFileName.0 = STRING: "myConfig.xml"
 AW2S-RMUv6-MIB::cfgFilePath.0 = STRING: "/etc/rmu/myConfig.xml"
```

Figure 35: RMU configuration file changed notification

Read-Only Variable	cfgDirName
SNMP OID	1.3.6.1.4.1.50488.2.20.3.0
Syntax	DirNameString
Description	RMU configuration files working directory. This is the directory where configuration files are saved.
CLI equiv.	cfg info -v

Read-Only Variable	cfgFileName
SNMP OID	1.3.6.1.4.1.50488.2.20.4.0
Syntax	FileNameString
Description	RMU configuration file name. The configuration file name may be changed by using the `cfgSave` command.
CLI equiv.	cfg info -v

Read-Only Variable	cfgFilePath
SNMP OID	1.3.6.1.4.1.50488.2.20.5.0
Syntax	FilePathString
Description	RMU configuration file absolute file path in the host's file system.
CLI equiv.	cfg info

Read-Write Command	cfgSave
SNMP OID	1.3.6.1.4.1.50488.2.20.6.0
Syntax	FileNameOrEmptyString
Description	Issuing a SET command on this field will cause the RMU to save its current configuration into a file using the string value of the SET command as file name.

	If the string value of the SET command is empty, then the RMU will save its configuration by backuping the current file in use then overwriting it. In case of 'Bad Value' failure of the SET command, a GET command will return an information string explaining the failure reason. Following a successful SET command, the value of this field is an empty string.
CLI equiv.	`cfg save [<fileName>] [-n]`

Notification	**cfgFileChanged**
SNMP OID	`1.3.6.1.4.1.50488.2.20.8`
Variables	`cfgFileName, cfgFilePath`
Description	This notification signifies that the current configuration file has changed. Usually following a write to `cfgSave`. The new configuration file name and file path are included in the notification.

12.5 RAT C&M server

The RAT C&M server provides services to RAT C&M clients.
The goal of the RAT C&M server is to answer to signal path configuration request coming in from the RAT C&M clients. The RMU then links the signal path configuration request to adequate radio equipment as desired, and informs the client how the signal path should be mapped across the fronthaul shared resources for correct operation.

Read-Only Variable	cmStatus
SNMP OID	1.3.6.1.4.1.50488.2.30.1.0
Syntax	RMUObjectStatus
Description	RAT C&M server status.
CLI equiv.	cm info

Read-Only Variable	cmStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.30.2.0
Syntax	InfoString
Description	RAT C&M server status information details. May be empty if no relevant information is available.
CLI equiv.	cm info

Notification	cmStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.30.3
Variables	cmStatus, cmStatusInfo
Description	This notification is emitted when the RAT C&M server's status has changed. The included values of cmStatus and cmStatusInfo provide the new status and possibly textual information regarding this new status.

When new RAT C&M clients are connected to the RMU, a service is registered for the new client. The cmServiceTable list all the currently registered RAT C&M clients.

Notification	cmServiceRegistered
SNMP OID	1.3.6.1.4.1.50488.2.30.12
Variables	cmServiceClientDescr, cmNumServices
Description	The cmServiceRegistered notification is emitted when a new RAT C&M client has been registered (connected) to the RMU RAT C&M server (a new service entry has been added to the cmServiceTable). The variable bindings include the new client's description cmServiceClientDescr. The cmNumServices is also updated accordingly.

Notification	cmServiceUnregistered
SNMP OID	1.3.6.1.4.1.50488.2.30.13
Variables	cmServiceClientDescr, cmNumServices

Description	The `cmServiceUnregistered` notification is emitted when a RAT C&M client has been unregistered (disconnected) from the RMU RAT C&M server (a service entry has been removed from the `cmServiceTable`). The variable bindings include the removed client's description `cmServiceClientDescr`. The `cmNumServices` is also updated accordingly.

Read-Only Variable	**cmNumServices**
SNMP OID	`1.3.6.1.4.1.50488.2.30.10.0`
Syntax	`Gauge32`
Description	Number of RAT C&M services managed by the RAT C&M server. Also corresponds to the number of service entries declared in the `cmServiceTable`.
CLI equiv.	`cm info -v`

Table	**cmServiceTable**
SNMP OID	`1.3.6.1.4.1.50488.2.30.11`
Description	List of RAT C&M service entries. The number of entries is given by the value of `cmNumServices`.

Each RAT C&M client may have one or more signal paths configuration requests (RAT C&M signal path). All the RAT C&M signal paths managed by the RAT C&M server are thus listed in the `cmPathTable`.

Figure 36 shows the RMU console output when an example Amarisoft LTEENB client is registered. Note here that errors appear due to the fact that no Radio Equipment is connected and thus the request cannot be fulfilled.

Figure 36: RMU console output on RAT C&M service registration

An extract of corresponding SNMP notifications received by the NMS is shown in Figure 37.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmServiceRegistered
 AW2S-RMUv6-MIB::cmServiceClientDescr.0 = STRING: ./lteenb-avx2 config/enb.cfg
 AW2S-RMUv6-MIB::cmNumServices.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathAdded
 AW2S-RMUv6-MIB::cmPathClientDescr.0.tx.0 = STRING: ./lteenb-avx2 config/enb.cfg
 AW2S-RMUv6-MIB::cmPathUuid.0.tx.0 = STRING: 0x00010000
 AW2S-RMUv6-MIB::cmNumPaths.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathStatusChanged
 AW2S-RMUv6-MIB::cmPathStatus.0.tx.0 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::cmPathStatusInfo.0.tx.0 = STRING: "Target radio equipment not connected"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathAdded
 AW2S-RMUv6-MIB::cmPathClientDescr.0.tx.1 = STRING: ./lteenb-avx2 config/enb.cfg
 AW2S-RMUv6-MIB::cmPathUuid.0.tx.1 = STRING: 0x00010100
 AW2S-RMUv6-MIB::cmNumPaths.0 = Gauge32: 2

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathStatusChanged
 AW2S-RMUv6-MIB::cmPathStatus.0.tx.1 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::cmPathStatusInfo.0.tx.1 = STRING: "Target radio equipment not connected"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathAdded
 AW2S-RMUv6-MIB::cmPathClientDescr.0.rx.0 = STRING: ./lteenb-avx2 config/enb.cfg
 AW2S-RMUv6-MIB::cmPathUuid.0.rx.0 = STRING: 0x00000000
 AW2S-RMUv6-MIB::cmNumPaths.0 = Gauge32: 3

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathStatusChanged
 AW2S-RMUv6-MIB::cmPathStatus.0.rx.0 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::cmPathStatusInfo.0.rx.0 = STRING: "Target radio equipment not connected"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathAdded
 AW2S-RMUv6-MIB::cmPathClientDescr.0.rx.1 = STRING: ./lteenb-avx2 config/enb.cfg
 AW2S-RMUv6-MIB::cmPathUuid.0.rx.1 = STRING: 0x00000100
 AW2S-RMUv6-MIB::cmNumPaths.0 = Gauge32: 4

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmPathStatusChanged
 AW2S-RMUv6-MIB::cmPathStatus.0.rx.1 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::cmPathStatusInfo.0.rx.1 = STRING: "Target radio equipment not connected"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (112210) 0:18:42.10
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::cmServiceStatusChanged
 AW2S-RMUv6-MIB::cmServiceStatus.0 = INTEGER: operational(4)
 AW2S-RMUv6-MIB::cmServiceStatusInfo.0 = ""
```

Figure 37: SNMP notifications following a RAT C&M service registration

Additionally, the `snmptable` application provides a user-friendly display of any SNMP table, Figure 38 demonstrates its usage on the `cmPathTable`.

```
user@shuttle:~$ snmptable -v 2c -c public -m ALL -Cb localhost cmPathTable
SNMP table: AW2S-RMUv6-MIB::cmPathTable

 ServiceIndex Direction Index  Status                                StatusInfo                 ClientDescr      Uuid              ReqMapping  ReqType
ReqDuplexMode     ReqPower ReqArfcn Enable  ProcessStep ProductName AntennaLabel CmdEnabled CmdPortId CmdStartbit CmdDelay CmdOffset MeasuredPwr
            0           tx        0 errored "Target radio equipment not connected" ./lteenb-avx2 config/enb.cfg 0x00010000 swa:0/port:0/node:0/ant:0 NR 20MHz
          tdd 3000 dBm/100   649942   true Node Mapping                                        false          0           0     0 ns  0 dB/100   0 dBm/100
            0           tx        1 errored "Target radio equipment not connected" ./lteenb-avx2 config/enb.cfg 0x00010100 swa:0/port:0/node:0/ant:1 NR 20MHz
          tdd 3000 dBm/100   649942   true Node Mapping                                        false          0           0     0 ns  0 dB/100   0 dBm/100
            0           rx        0 errored "Target radio equipment not connected" ./lteenb-avx2 config/enb.cfg 0x00000000 swa:0/port:0/node:0/ant:0 NR 20MHz
          tdd    0 dBm/100   649942   true Node Mapping                                        false          0           0     0 ns  0 dB/100   0 dBm/100
            0           rx        1 errored "Target radio equipment not connected" ./lteenb-avx2 config/enb.cfg 0x00000100 swa:0/port:0/node:0/ant:1 NR 20MHz
          tdd    0 dBm/100   649942   true Node Mapping                                        false          0           0     0 ns  0 dB/100   0 dBm/100
user@shuttle:~$
```

Figure 38: Table display of all RAT C&M signal path configuration requests

Notification	cmPathAdded
SNMP OID	1.3.6.1.4.1.50488.2.30.22
Variables	cmPathClientDescr, cmPathUuid, cmNumPaths
Description	The cmPathAdded notification is emitted when a new RAT C&M signal path configuration request has been received by the RMU (a new path entry has been added to the cmPathTable). The variable bindings include the parent client's description cmPathClientDescr and the signal path's user-specified cmPathUuid. The cmNumPaths is also updated accordingly.

Notification	cmPathRemoved
SNMP OID	1.3.6.1.4.1.50488.2.30.23
Variables	cmPathClientDescr, cmPathUuid, cmNumPaths
Description	The cmPathRemoved notification is emitted when a RAT C&M signal path configuration request has been removed from the RMU (a path entry has been removed from the cmPathTable). The variable bindings include the parent client's description cmPathClientDescr and the signal path's user-specified cmPathUuid. The cmNumPaths is also updated accordingly.

Read-Only Variable	cmNumPaths
SNMP OID	1.3.6.1.4.1.50488.2.30.20.0
Syntax	Gauge32
Description	Number of RAT C&M signal paths managed by the RAT C&M server. Also corresponds to the number of path entries declared in the cmPathTable.

Table	cmPathTable
SNMP OID	1.3.6.1.4.1.50488.2.30.21
Description	List of RAT C&M signal path request entries. The number of entries is given by the value of cmNumPaths.

12.5.1 RAT C&M client (service) entry

RAT C&M clients can be managed individually. Each entry provides some information elements such as the status and the number of requested signal path configurations for each client.

Table entry	cmServiceEntry
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1
Indexes	cmServiceIndex
Description	RAT C&M service entry containing management information applicable to a RAT C&M service.

Read-Only Variable	cmServiceIndex
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.1.cmServiceIndex(X)
Syntax	Gauge32
Description	Unique value for each RAT C&M service used to index the list of RAT C&M service entries.

Read-Only Variable	cmStatus
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.2.cmServiceIndex(X)
Syntax	RMUObjectStatus
Description	RAT C&M service status.
CLI equiv.	cm/service:X info

Read-Only Variable	cmStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.3.cmServiceIndex(X)
Syntax	InfoString
Description	RAT C&M service status information details. May be empty if no relevant information is available.
CLI equiv.	cm/service:X info

Notification	cmServiceStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.30.14
Variables	cmServiceStatus, cmServiceStatusInfo
Description	This notification is emitted when a RAT C&M client service's status has changed. The included values of cmServiceStatus and cmServiceStatusInfo provide the new status and possibly textual information regarding this new status.

Read-Only Variable	cmServiceClientDescr
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.4.cmServiceIndex(X)
Syntax	DisplayString
Description	The name of the RAT C&M service client as provided by the client application through the RAT C&M API.
CLI equiv.	cm/service:X info

Read-Only Variable	cmServiceNumReqTxPaths
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.5.cmServiceIndex(X)
Syntax	Gauge32
Description	Number of client-side TX signal paths configurations requested for the service.
CLI equiv.	cm/service:X info

Read-Only Variable	cmServiceNumReqRxPaths
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.6.cmServiceIndex(X)
Syntax	Gauge32
Description	Number of client-side RX signal paths configurations requested for the service.
CLI equiv.	cm/service:X info

The only writable variable for a RAT C&M service is the enablement. This means that a specific client can be enabled/disabled dynamically as desired by the NMS. By default, services are enabled when registered.
Signal path configuration requests for a given RAT C&M client are managed only when the service for this client is enabled.

Read-Write Variable	cmServiceEnable
SNMP OID	1.3.6.1.4.1.50488.2.30.11.1.7.cmServiceIndex(X)
Syntax	TruthValue
Description	Service enablement control. When set to true(1), the service is enabled and client-side signal paths configuration requests for this service are handled by the RAT C&M server. When set to false(2), the service is not managed, causing client-side signal paths configuration requests to be ignored: a disable command is forwarded to the client for all of this service's signal path configuration requests.
CLI equiv.	cm/service:X enabled [true|false]

12.5.2 RAT C&M signal path configuration request entry

Moreover, each RAT C&M client signal path configuration request can be monitored individually. Each signal path configuration request entry provides some information elements such as the status of the signal path, the actual request parameters from the RAT C&M client, as well as the linked radio equipment.

Table entry	cmPathEntry
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1
Indexes	cmPathServiceIndex, cmPathDirection, cmPathIndex
Description	RAT C&M path entry containing management information applicable to a RAT C&M signal path request.

Read-Only Variable	cmPathServiceIndex
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.1.cmPathServiceIndex(X) .cmPathDirection.cmPathIndex(Y)
Syntax	Gauge32
Description	Index of the parent RAT C&M service for which this signal path configuration request is coming from, used to index the list of RAT C&M signal path request entries.

Read-Only Variable	cmPathDirection
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.2.cmPathServiceIndex(X) .cmPathDirection.cmPathIndex(Y)
Syntax	INTEGER (enumerated)

Description	Indicates the direction of the RAT C&M signal path request, where `tx(1)` means the signal is sourced from the RAT stack and is to be mapped to the RF antenna port, and `rx(2)` means the signal is mapped from the RF antenna port and terminated by the RAT stack. The `cmPathDirection` value is also used to index the list of RAT C&M signal path request entries.

Read-Only Variable	**cmPathIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.30.21.1.3.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)`
Syntax	`Gauge32`
Description	Unique value for each signal path request for a given `cmPathServiceIndex` and `cmPathDirection`, used to index the list of RAT C&M signal path request entries.

Read-Only Variable	**cmPathStatus**
SNMP OID	`1.3.6.1.4.1.50488.2.30.21.1.4.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)`
Syntax	`RMUObjectStatus`
Description	RAT C&M signal path status.
CLI equiv.	`cm/service:X/tx:Y info` `cm/service:X/rx:Y info`

Read-Only Variable	**cmPathStatusInfo**
SNMP OID	`1.3.6.1.4.1.50488.2.30.21.1.5.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)`
Syntax	`InfoString`
Description	RAT C&M signal path status information details. May be empty if no relevant information is available.
CLI equiv.	`cm/service:X/tx:Y info` `cm/service:X/rx:Y info`

Notification	**cmPathStatusChanged**
SNMP OID	`1.3.6.1.4.1.50488.2.30.24`
Variables	`cmPathStatus, cmPathStatusInfo`
Description	This notification is emitted when a RAT C&M signal path configuration request handling status has changed. The included values of `cmPathStatus` and `cmPathStatusInfo` provide the new status and possibly textual information regarding this new status.

The following read-only elements are the values sent by the RAT C&M client for this signal path configuration request. They define how the signal path are to be integrated within the fronthaul.

Read-Only Variable	cmPathClientDescr
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.6.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	The name of the RAT C&M service client, for this signal path request, as provided by the client application through the RAT C&M API.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathUuid
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.7.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	User unique identifier for this signal path request, as provided by the client application through the RAT C&M API.
CLI equiv.	cm/service:X/tx:Y info -v cm/service:X/rx:Y info -v

Read-Only Variable	cmPathReqMapping
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.8.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	Requested fronthaul/antenna mapping for this signal path request, as provided by the client application through the RAT C&M API. The string is in the form 'swa:X/port:Y/node:Z/ant:U' where X is the Swallow board identifier, Y the Swallow port identifier, Z the number of networking radio equipments, and U the antenna port identifier.
CLI equiv.	cm/service:X/tx:Y info -v cm/service:X/rx:Y info -v

Read-Only Variable	cmPathReqType
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.9.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	A string describing the signal path request type, such as technology (LTE or NR) and channel bandwidth, as provided by the client application through the RAT C&M API.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathReqDuplexMode
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.10.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	INTEGER (enumerated)
Description	The signal path request duplex mode, as provided by the client application through the RAT C&M API. When set to fdd(1), the signal path operates in Frequency Division Duplexing. When set to tdd(2), the signal path operates in Time Division Duplexing.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathReqPower
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.11.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	Integer32 (dBm/100)
Description	The signal path request maximum output power, as provided by the client application through the RAT C&M API. Only applicable for TX signal paths (cmPathDirection is tx(1)).
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathReqArfcn
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.12.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	Gauge32
Description	The signal path request Absolute Radio Frequency Channel Number, as provided by the client application through the RAT C&M API. Defines the RF center frequency of the signal path.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Each RAT C&M signal path can be enabled or disabled individually. By default, signal paths are enabled.
When a signal path is disabled, it is unlinked from any radio equipment that it may have been linked to, and a disable command is sent by the RAT C&M server to the corresponding RAT C&M service, causing the I/Q data to be blanked for this signal path. Figure 39 shows the command for a user-triggered disablement. The result is visible in the Amarisoft LTEENB console, displayed in Figure 40.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost cmPathEnable.0.1.1 i false
Unexpected index type: 7 cmPathServiceIndex 0.1.1
AW2S-RMUv6-MIB::cmPathEnable.0.tx.1 = INTEGER: false(2)
user@shuttle:~$
```

Figure 39: Disabling a RAT C&M signal path

```
(enb) [CELL0_TX1] not operational: Disabled by user
```

Figure 40: Amarisoft LTEENB console output on signal path disablement

Read-Write Variable	cmPathEnable
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.13.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	TruthValue
Description	RAT C&M signal path enablement control. When set to true(1), the signal path request is managed and the RMU will attempt to link this signal path configuration request to a suitable equipment for operation. When set to false(2), this signal path configuration request is not managed by the RMU and is thus ignored: a disable command is forwarded to the client for this signal path.
CLI equiv.	cm/service:X/tx:Y enabled [true|false] cm/service:X/rx:Y enabled [true|false]

Finally, the following read-only (informative) variables display the radio equipment on which the signal path is mapped, as well as the C&M command content sent back to the RAT C&M client..

Read-Only Variable	cmPathProcessStep
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.14.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	String describing the current C&M signal path request state machine step.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathProductName
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.15.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	Product name of the node on which the RAT C&M signal path is mapped. The string is a concatenation of the product part number, revision and serial number An empty string means that the signal path is not yet mapped (e.g. the target Radio Equipment is not managed by the RMU).
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathAntennaLabel
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.16.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	DisplayString
Description	Name of the physical antenna port on which the RAT C&M signal path is mapped, as indicated on the radio equipment housing. An empty string means that the signal path is not yet mapped (e.g. the target antenna port is unavailable or disabled).
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathCmdEnabled
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.17.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	TruthValue
Description	Enablement value of the RAT C&M command sent back to the client application. If true(1), then it means that the RF signal path is fully allocated and the client application shall proceed with transmission of I/Q data.
CLI equiv.	cm/service:X/tx:Y info cm/service:X/rx:Y info

Read-Only Variable	cmPathCmdPortId
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.18.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	Gauge32
Description	Port identifier value of the RAT C&M command sent back to the client application. This value defines across which Swallow CPRI port the I/Q data is to be transmitted.
CLI equiv.	cm/service:X/tx:Y info -v cm/service:X/rx:Y info -v

Read-Only Variable	cmPathCmdStartbit
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.19.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	Gauge32
Description	CPRI frame startbit value of the RAT C&M command sent back to the client application. Defines where the AxC container is allocated in the CPRI basic frame for this signal path.
CLI equiv.	cm/service:X/tx:Y info -v cm/service:X/rx:Y info -v

Read-Only Variable	cmPathCmdDelay
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.20.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)
Syntax	Gauge32 (ns)
Description	Signal path delay value of the RAT C&M command sent back to the client application. This value corresponds to the sum of the fronthaul delay (cable+networking delay) and radio equipment latency (T2A/TA3). Used by the client application for delay compensation.
CLI equiv.	cm/service:X/tx:Y info -v cm/service:X/rx:Y info -v

Read-Only Variable	cmPathCmdOffset
SNMP OID	1.3.6.1.4.1.50488.2.30.21.1.21.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)

Syntax	`Integer32` (dB/100)
Description	Rx conversion gain value of the RAT C&M command sent back to the client application. This value corresponds to the Rx dBFS to dBm conversion gain of the radio equipment on which the signal path is mapped. Used by the client application to estimate Rx RF power levels from digital power levels.
CLI equiv.	`cm/service:X/tx:Y info -v` `cm/service:X/rx:Y info -v`

Read-Only Variable	**cmPathMeasuredPwr**
SNMP OID	`1.3.6.1.4.1.50488.2.30.21.1.22.cmPathServiceIndex(X).cmPathDirection.cmPathIndex(Y)`
Syntax	`Integer32` (dBm/100)
Description	Measured power for this signal path. The power measurement is done by the radio equipment as the integrated whole-band power within a 10 milliseconds window. This parameter is updated every 30 seconds. If the signal path object is not created on the radio equipment, then the value is invalid.
CLI equiv.	`cm/service:X/tx:Y info -v` `cm/service:X/rx:Y info -v`

12.6 Swallow boards management

The Swallow boards management relates to all management parameters associated to Swallow PCI-Express boards. The RMU can support multiple Swallow boards plugged into the host computer.

Read-Only Variable	numSwallow
SNMP OID	1.3.6.1.4.1.50488.2.40.10.0
Syntax	Gauge32
Description	Number of Swallow boards populated in the system and managed by the RMU.

Table	swallowTable
SNMP OID	1.3.6.1.4.1.50488.2.40.11
Description	List of Swallow board entries. The number of entries is given by the value of numSwallow.

Each Swallow board can populate multiple CPRI ports, which are listed in the swallowPortTable, as shown in Figure 41.

Figure 41: Swallow CPRI master port table display

Read-Only Variable	numSwallowPort
SNMP OID	1.3.6.1.4.1.50488.2.40.20.0
Syntax	Gauge32
Description	Total number of fronthaul (CPRI) ports managed by the RMU. Each Swallow board can populate multiple CPRI ports.

Table	swallowPortTable
SNMP OID	1.3.6.1.4.1.50488.2.40.21
Description	List of Swallow port entries. The number of entries is given by the value of numSwallowPort.

12.6.1 Swallow board entry

The Swallow board management provides elements relative to a Swallow hardware device. The principal managed features are the device versioning, time/frequency synchronization setup, and GPS information.

Table entry	swallowEntry
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1
Indexes	swallowIndex
Description	Swallow board entry containing management information applicable to a Swallow board.

Read-Only Variable	swallowIndex
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.1.swallowIndex(X)
Syntax	Gauge32
Description	Unique value for each Swallow board used to index the list of Swallow board entries. This is the device 'minor' identifier.

Read-Only Variable	swallowStatus
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.2.swallowIndex(X)
Syntax	RMUObjectStatus
Description	Swallow board status.
CLI equiv.	swa:X info

Read-Only Variable	swallowStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.3.swallowIndex(X)
Syntax	InfoString
Description	Swallow board status information details. May be empty if no relevant information is available.
CLI equiv.	swa:X info

Notification	swallowStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.40.12
Variables	swallowStatus, swallowStatusInfo
Description	This notification is emitted when the Swallow board's status has changed. The included values of swallowStatus and swallowStatusInfo provide the new status and possibly textual information regarding this new status.

Swallow board versioning information is readily available, as shown in Figure 42.

```
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost swallowHwType.0 swallowHwRev.0 swallowFwVersion.0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
AW2S-RMUv6-MIB::swallowHwType.0 = STRING: Type B (Swallow V1.5)
AW2S-RMUv6-MIB::swallowHwRev.0 = Gauge32: 1
AW2S-RMUv6-MIB::swallowFwVersion.0 = STRING: Type B (Swallow V1.5) FW6.7 (V6B7_3p_3c444.bit)
user@shuttle:~$
```

Figure 42: Swallow device version

Read-Only Variable	swallowHwType
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.4.swallowIndex(X)
Syntax	DisplayString
Description	Descriptive string of the Swallow board hardware type.

Read-Only Variable	swallowHwRev
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.5.swallowIndex(X)
Syntax	Gauge32
Description	Swallow board hardware revision code.

Read-Only Variable	swallowFwVersion
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.6.swallowIndex(X)
Syntax	DisplayString
Description	Descriptive string of the Swallow board firmware type and version.
CLI equiv.	swa:X info

Read-Only Variable	swallowTemperature
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.7.swallowIndex(X)
Syntax	Integer32 (degC)
Description	Junction temperature of the Swallow board FPGA.
CLI equiv.	swa:X info

The Swallow device time and frequency synchronization is tightly coupled to the Swallow object status.

```
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost swallowSyncMode.0 swallowSyncResult.0 swallowStatus.0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
AW2S-RMUv6-MIB::swallowSyncMode.0 = INTEGER: gps(2)
AW2S-RMUv6-MIB::swallowSyncResult.0 = INTEGER: gpsPpsMissing(3)
AW2S-RMUv6-MIB::swallowStatus.0 = INTEGER: preOperational(3)
```

Figure 43: Retrieving Swallow synchronization result

Changing the synchronization mode may affect the device status. Additionally, SNMP notifications are emitted on synchronization result change, as shown in Figure 45.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost swallowSyncMode.0 i freerun
Unexpected index type: 7 swallowIndex 0
AW2S-RMUv6-MIB::swallowSyncMode.0 = INTEGER: freerun(0)
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost swallowSyncMode.0 swallowSyncResult.0 swallowStatus.0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
Unexpected index type: 7 swallowIndex 0
AW2S-RMUv6-MIB::swallowSyncMode.0 = INTEGER: freerun(0)
AW2S-RMUv6-MIB::swallowSyncResult.0 = INTEGER: ok(0)
AW2S-RMUv6-MIB::swallowStatus.0 = INTEGER: operational(4)
user@shuttle:~$
```

Figure 44: Changing the Swallow synchronization scheme

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (212743) 0:35:27.43
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowSyncResultChanged
 AW2S-RMUv6-MIB::swallowSyncMode.0 = INTEGER: freerun(0)
 AW2S-RMUv6-MIB::swallowSyncResult.0 = INTEGER: ok(0)

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (212743) 0:35:27.43
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowStatusChanged
 AW2S-RMUv6-MIB::swallowStatus.0 = INTEGER: operational(4)
 AW2S-RMUv6-MIB::swallowStatusInfo.0 = STRING: "Sync mode Internal reference: Synchronized"
```

Figure 45: Notification on Swallow synchronization result change event

Read-Write Variable	**swallowSyncMode**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.8.swallowIndex(X)
Syntax	INTEGER (enumerated)
Description	Sets the current external time and/or frequency synchronization method used by the Swallow board. ➢ freerun(0) : Internal clock reference is used. The board runs on its internal clock oscillator and is not synchronized to any external source. ➢ refclk(1) : The board synchronizes itself on an external 15.36 MHz reference clock, providing frequency synchronization. This mode is not valid for type D, E and F boards. ➢ gps(2) : The board synchronizes itself on the GPS system, providing time and frequency synchronization. GPS synchronization requires the usage of a GPS antenna. ➢ umts(3) : The board synchronizes itself on an external 100 Hz UMTS synchronization pulse, providing time and frequency synchronization. This mode is not valid for type D, E and F boards. ➢ pps-rise(4) : The board synchronizes itself on the rising edge of an external PPS synchronization pulse, providing time and frequency synchronization. This mode is not valid for type D, E and F boards. ➢ pps-fall(5) : The board synchronizes itself on the falling edge of an external PPS synchronization pulse, providing time and frequency synchronization. This mode is not valid for type D, E and F boards.
CLI equiv.	swa:X sync-mode [freerun|refclk|gps|umts|pps-rise|pps-fall]

Read-Only Variable	**swallowSyncResult**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.9.swallowIndex(X)
Syntax	INTEGER (enumerated)
Description	Describes the current status of the synchronization method used by the board. ➢ ok(0) : Synchronization has been achieved. When using external synchronization methods, indicates that the board has fully locked onto the external synchronization source. ➢ pllUnlocked(1) : Indicates that the board's Phase-Locked Loop is unlocked. This value means that the board's hardware may be defective, and implies CPRI link issues as well as abnormal RF performances. ➢ extRefMissing(2) : Applicable only when swallowSyncMode is set to refclk(1). Indicates that an external 15.36 MHz reference clock is not detected. This value may imply CPRI link issues as well as abnormal RF performances.

	➢ `gpsPpsMissing(3)` : Applicable only when `swallowSyncMode` is set to `gps(2)`, `pps-rise(4)` or `pps-fall(5)`. Indicates that no PPS (Pulse-Per-Second) signal is received, meaning that either the GPS receiver has not locked onto the satellite constellation, or that the external synchronization pulse is not detected, so no timing information can be retrieved. In this state, the board still runs on its internal clock oscillator so CPRI link and RF performances are still maintained. ➢ `umtsMissing(4)` : Applicable only when `swallowSyncMode` is set to `umts(3)`. Indicates that an external 100 Hz synchronization pulse is not detected so no timing information can be retrieved. In this state, the board still runs on its internal clock oscillator so CPRI link and RF performances are still maintained. ➢ `inProgress(5)` : Applicable only when `swallowSyncMode` is set to either `gps(2)`, `umts(3)`,`pps-rise(4)` or `pps-fall(5)`. The external synchronization signal has been detected and the board is in the process of locking itself onto it. This state may last a few minutes before `swallowSyncResult` becomes `ok(0)`. In this state, CPRI link and RF performances are still maintained. ➢ `unknown(6)` : Unknown synchronization status.
CLI equiv.	`swa:X info`

Notification	**swallowSyncResultChanged**
SNMP OID	`1.3.6.1.4.1.50488.2.40.13`
Variables	`swallowSyncMode, swallowSyncResult`
Description	This notification is emitted when current status of the synchronization method used by the Swallow board has changed. The included values of `swallowSyncMode` and `swallowSyncResult` provide information regarding the current synchronization mode and associated synchronization status (result).

Read-Write Variable	**swallowSyncOutput**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.10.swallowIndex(X)`
Syntax	`INTEGER` (enumerated)
Description	Sets the synchronization signal output issued by the Swallow board. Synchronization output signals are not supported by type D, E and F boards. ➢ `refclk(0)` : Signal output is a 15.36 MHz reference clock. ➢ `umts(1)` : Signal output is a 100 Hz UMTS synchronization pulse, time-aligned with regards to the external input source if applicable.
CLI equiv.	`swa:X sync-output [refclk|umts]`

Read-Only Variable	**swallowSyncClockOffset**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.11.swallowIndex(X)`
Syntax	`Integer32`
Description	Number of accumulated clock cycles error (clock deviation) from the external synchronization source since the beginning of the synchronization process for the Swallow board. A value oscillating around 0 +/- 50 indicates that the board has locked onto the external synchronization source (e.g. GPS PPS or UMTS pulse).

	The timing error (board time offset compared to the external synchronization source) can be approximated by multiplying the value by 5.4 nanoseconds. For instance, a value of 20 indicates that the board internal time is offset by around 108 nanoseconds compared to the external synchronization source.
CLI equiv.	`swa:X info -v`

Read-Only Variable	**swallowNumPorts**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.12.swallowIndex(X)
Syntax	Gauge32
Description	Number of fronthaul (CPRI) ports available for this Swallow board.

12.6.1.1 GPS NMEA parser

The RMU offers an integrated GPS NMEA parser for each Swallow board. It decodes GGA sentences received from the GPS receiver. The following variables allows monitoring of GPS fix and time information from the NMS. Also, see Integrated GPS receiver raw NMEA data for more information.

Read-Only Variable	**swallowGpsStatus**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.20.swallowIndex(X)
Syntax	RMUObjectStatus
Description	GPS NMEA parser status.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsStatusInfo**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.21.swallowIndex(X)
Syntax	InfoString
Description	GPS NMEA parser status information details. May be empty if no relevant information is available.
CLI equiv.	`swa:X/gps info`

Notification	**swallowGpsStatusChanged**
SNMP OID	1.3.6.1.4.1.50488.2.40.14
Variables	swallowGpsStatus, swallowGpsStatusInfo
Description	This notification is emitted when the GPS NMEA parser status has changed. The included values of swallowGpsStatus and swallowGpsStatusInfo provide the new status and possibly textual information regarding this new status.

Read-Only Variable	**swallowGpsFixDataAge**
SNMP OID	1.3.6.1.4.1.50488.2.40.11.1.22.swallowIndex(X)
Syntax	TimeTicks
Description	The time (in hundredths of a second) since the last valid GPS fix was received.

	Reads zero if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsSentenceId**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.23.swallowIndex(X)`
Syntax	`DisplayString`
Description	NMEA sentence identifier for which the GPS fix was received. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsReportUTC**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.24.swallowIndex(X)`
Syntax	`TimeTicks`
Description	Latest GPS fix data UTC report time (in hundredths of a second). Reads zero if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsLatitude**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.25.swallowIndex(X)`
Syntax	`DisplayString`
Description	Latest GPS fix reported latitude in decimal degrees format. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsLongitude**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.26.swallowIndex(X)`
Syntax	`DisplayString`
Description	Latest GPS fix reported longitude in decimal degrees format. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsFixQuality**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.27.swallowIndex(X)`
Syntax	`Gauge32`
Description	Latest GPS fix quality indicator as received in GGA fix data, where the value 0 means either invalid fix or no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsNumSatellites**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.28.swallowIndex(X)`
Syntax	`Gauge32`

Description	Latest GPS fix number of satellites in view. Reads zero if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsHDOP**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.29.swallowIndex(X)`
Syntax	`DisplayString`
Description	Latest GPS fix horizontal dilution of precision. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsAltitude**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.30.swallowIndex(X)`
Syntax	`DisplayString`
Description	Latest GPS fix altitude above mean sea level. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

Read-Only Variable	**swallowGpsGeoidHeight**
SNMP OID	`1.3.6.1.4.1.50488.2.40.11.1.31.swallowIndex(X)`
Syntax	`DisplayString`
Description	Latest GPS fix height of geoid (mean sea level) above WGS84 ellipsoid. Empty string if no GPS fix has been received yet.
CLI equiv.	`swa:X/gps info`

12.6.2 Swallow CPRI port entry

A Swallow board typically provides one or more CPRI master port. Each CPRI port entry can be managed individually and configures how the CPRI port operates.

Table entry	**swallowPortEntry**
SNMP OID	`1.3.6.1.4.1.50488.2.40.21.1`
Indexes	`swallowPortBoardIndex, swallowPortIndex`
Description	Swallow port entry containing management information applicable to a Swallow fronthaul (CPRI) port/interface.

Read-Only Variable	**swallowPortBoardIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.40.21.1.1.swallowPortBoardIndex(X)` `.swallowPortIndex(Y)`
Syntax	`Gauge32`
Description	Device '`minor`' identifier of the parent Swallow board for this Swallow port.

Read-Only Variable	swallowPortIndex
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.2.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	Gauge32
Description	Unique value for each Swallow port, used to index the list of Swallow port entries. This value also corresponds to the physically labeled index of the port (e.g. SFP_1 if swallowPortIndex is 1).

Read-Only Variable	swallowPortStatus
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.3.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	RMUObjectStatus
Description	Swallow port status.
CLI equiv.	swa:X/port:Y info

Read-Only Variable	swallowPortStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.4.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	InfoString
Description	Swallow port status information details. May be empty if no relevant information is available.
CLI equiv.	swa:X/port:Y info

Notification	swallowPortStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.40.22
Variables	swallowPortStatus, swallowPortStatusInfo
Description	This notification is emitted when a Swallow port's status has changed. The included values of swallowPortStatus and swallowPortStatusInfo provide the new status and possibly textual information regarding this new status.

Read-Write Variable	swallowPortLineSpeed
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.5.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	INTEGER (enumerated)
Description	Sets the CPRI line speed option to be used for this CPRI port. The line speed options are defined by the CPRI specification as follow: ➢ disabled(0) : CPRI port is disabled, SFP transmission is disabled. ➢ option1(1) : CPRI port is enabled at 614.4 Mbps line bit rate. This option is not allowed by the RMU. ➢ option2(2) : CPRI port is enabled at 1228.8 Mbps line bit rate. This option is not allowed by the RMU. ➢ option3(3) : CPRI port is enabled at 2457.6 Mbps line bit rate. ➢ option4(4) : CPRI port is enabled at 3072.0 Mbps line bit rate.

	➢ `option5(5)` : CPRI port is enabled at 4915.2 Mbps line bit rate. ➢ `option6(6)` : CPRI port is enabled at 6144 Mbps line bit rate. ➢ `option7(7)` : CPRI port is enabled at 9830.4 Mbps line bit rate. This option is not supported by type A boards. ➢ `option8(8)` : CPRI port is enabled at 10137.6 Mbps line bit rate. This option is not supported by type A boards. Changing the Swallow port mode will cause all Radio Equipment nodes currently connected to this CPRI port to be stopped/disconnected from the RMU momentarily.
CLI equiv.	`swa:X/port:Y line-speed [<option>]`

Read-Write Variable	**swallowPortDefragmentAllow**
SNMP OID	`1.3.6.1.4.1.50488.2.40.21.1.6.swallowPortBoardIndex(X).swallowPortIndex(Y)`
Syntax	`TruthValue`
Description	When a new RAT C&M signal path request is received by the RMU, the allocation of the signal path on the fronthaul interface requires the usage of part of the CPRI basic frame for I/Q data plane transport. When signal paths are added and removed, the CPRI basic frame may get fragmented over time and the need to defragment may rise in order to further allow CPRI basic frame signal path allocation for new signal paths. However, the defragmentation of the CPRI basic frame may imply short service interruption for one or more signal paths as the allocation of those signal paths may get reshuffled dynamically. The value sets whether defragmentation of the CPRI basic frame is allowed. When set to `true(1)`, the RMU will defragment the CPRI basic frame if it detects that it is fragmented and a defragmentation will allow the allocation of more signal paths. This configuration guarantees optimal CPRI basic frame usage but service may be interrupted when a new signal path request allocation is received. When set to `false(2)`, the RMU never defragments the CPRI basic frame. This setting guarantees no service interruption, but the allocator may fail to allocate a new signal path if the CPRI basic frame is fragmented.
CLI equiv.	`swa:X/port:Y defragment-allowed [true|false]`

Read-Only Variable	**swallowPortRole**
SNMP OID	`1.3.6.1.4.1.50488.2.40.21.1.7.swallowPortBoardIndex(X).swallowPortIndex(Y)`
Syntax	`INTEGER` (enumerated)
Description	Role of the Swallow CPRI port. When the value is `master(1)`, then the port is operating as a CPRI master. Where `slave(2)`, the port is operating as a CPRI slave. The role may change during runtime depending on the link partner's (radio equipment CPRI port) role.
CLI equiv.	`swa:X/port:Y info`

Read-Only Variable	swallowPortLinkIsUp
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.8.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	TruthValue
Description	This value indicates whether the CPRI link state for this CPRI port is either UP (true(1)) or DOWN (false(2)). A CPRI link UP state indicates that the SFP link is operational, but does not necessarily means that the link partner is fully managed by the RMU already (e.g. ORI TCP connection is possibly not yet achieved).
CLI equiv.	swa:X/port:Y info

Notification	swallowPortLinkUp
SNMP OID	1.3.6.1.4.1.50488.2.40.23
Variables	swallowPortRole, swallowPortLineSpeed
Description	This notification is emitted when the CPRI link state of a CPRI port has changed from DOWN to UP, meaning that the SFP link has been achieved. The included values of swallowPortRole and swallowPortLineSpeed indicate the operating CPRI port role and line speed option selected.

Notification	swallowPortLinkDown
SNMP OID	1.3.6.1.4.1.50488.2.40.24
Variables	swallowPortRole, swallowPortLineSpeed
Description	This notification is emitted when the CPRI link state of a CPRI port has changed from UP to DOWN, meaning that the SFP link has been lost. The included values of swallowPortRole and swallowPortLineSpeed indicate the operating CPRI port role and line speed option selected.

Read-Only Variable	swallowPortTxFrameRemSize
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.9.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	Gauge32
Description	Number of bits remaining and available for allocation in the CPRI Tx basic frame.

Notification	swallowPortTxFrameDefragmenting
SNMP OID	1.3.6.1.4.1.50488.2.40.25
Variables	swallowPortTxFrameRemSize
Description	This notification is triggered when the CPRI Tx basic frame defragmentation algorithm is started. Defragmentation usually requires the momentary disablement of one or more Tx signal paths in order to optimize fronthaul resource sharing, inducing loss of service for a couple of seconds at most. This event can only happen if swallowPortDefragmentAllow is true(1) when adding or enabling new Tx signal paths.

Read-Only Variable	swallowPortRxFrameRemSize
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.10.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	Gauge32
Description	Number of bits remaining and available for allocation in the CPRI Rx basic frame.

Notification	swallowPortRxFrameDefragmenting
SNMP OID	1.3.6.1.4.1.50488.2.40.26
Variables	swallowPortTxFrameRemSize
Description	This notification is triggered when the CPRI Rx basic frame defragmentation algorithm is started. Defragmentation usually requires the momentary disablement of one or more Rx signal paths in order to optimize fronthaul resource sharing, inducing loss of service for a couple of seconds at most. This event can only happen if swallowPortDefragmentAllow is true(1) when adding or enabling new Rx signal paths.

Read-Only Variable	swallowPortNumNodes
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.14.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	Gauge32
Description	Number of nodes (Radio Equipments) connected to this CPRI port. A connected node is defined as a node for which the ORI TCP link is fully established and for which the alignment procedure has completed.

12.6.2.1 DHCP parser

The DHCP parser, which operates on a given Swallow CPRI Master Port, parses incoming DHCP requests and extracts ORI-related parameters from the Ethernet over CPRI transport layer. Those extracted information elements (such as IP address, ORI TCP Port, RE vendor ID) are used to establish the ORI connection with Radio Equipments. If the DHCP parser is in error state, then new Radio Equipments cannot be linked to the RMU.

Read-Only Variable	swallowPortDhcpStatus
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.11.swallowPortBoardIndex(X).swallowPortIndex(Y)
Syntax	RMUObjectStatus
Description	Swallow port DHCP parser status.
CLI equiv.	swa:X/port:Y/dhcp info

Read-Only Variable	swallowPortDhcpStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.40.21.1.12.swallowPortBoardIndex(X).swallowPortIndex(Y)

Syntax	`InfoString`
Description	Swallow port DHCP parser status information details. May be empty if no relevant information is available.
CLI equiv.	`swa:X/port:Y/dhcp info`

Notification	**swallowPortDhcpStatusChanged**
SNMP OID	`1.3.6.1.4.1.50488.2.40.27`
Variables	`swallowPortDhcpStatus, swallowPortDhcpStatusInfo`
Description	This notification is emitted when a Swallow port DHCP parser's status has changed. The included values of `swallowPortDhcpStatus` and `swallowPortDhcpStatusInfo` provide the new status and possibly textual information regarding this new status.

Read-Only Variable	**swallowPortDhcpIfName**
SNMP OID	`1.3.6.1.4.1.50488.2.40.21.1.13.swallowPortBoardIndex(X).swallowPortIndex(Y)`
Syntax	`DisplayString`
Description	OS-name of the Ethernet Over CPRI network interface on which the DHCP parser for this Swallow port is running.

12.7 Radio interface

The radio interface concerns all management parameters that relates to the Radio Equipments (RRH). In the Swallow V6 RMU, a Radio Equipment is also named a Node (those terms are interchangeable), and realizes the operation of transmitting or receiving RF power on its antenna connectors.

12.7.1 Radio Equipment control & management

Radio Equipments become connected to the RMU when the ORI TCP link is fully established between the RMU and the remote node. When connected, the node is fully under the supervision of the RMU for control & management. The RMU will then allocate and configure RF signal paths on the node, according to the system's parameters. Moreover, the RMU implements node (topological) identification, health monitoring, product versioning, fault management, and other features, according to the specification of the Open Radio Interface. Under this protocol, the RMU acts as the Radio Equipment Controller (REC) while the node is the Radio Equipment (RE).

The `nodeTable` list all the currently connected Radio Equipments.

Notification	**nodeConnected**
SNMP OID	1.3.6.1.4.1.50488.2.50.12
Variables	nodeProductName, nodeHardwareVer, numNodes
Description	The nodeConnected notification is emitted when a new node (Radio Equipment) is connected and becomes managed by the RMU. A connected node is defined as a node for which the ORI TCP link is fully established and for which the alignment procedure has completed. Variable bindings nodeProductName and nodeHardwareVer provide product identification. A new node entry is added to the nodeTable and the numNodes variable is updated accordingly.

Notification	**nodeDisconnected**
SNMP OID	1.3.6.1.4.1.50488.2.50.13
Variables	nodeProductName, nodeHardwareVer, numNodes
Description	The nodeDisonnected notification is emitted when a node (Radio Equipment) is disconnected from the RMU. A node disconnection event usually means that fiber link was lost, or that the Radio Equipment was restarted (i.e. after a nodeReset command), or that some other error caused the ORI TCP link to fail. Variable bindings nodeProductName and nodeHardwareVer provide product identification. The numNodes is also updated accordingly.

Read-Only Variable	**numNodes**
SNMP OID	1.3.6.1.4.1.50488.2.50.10.0
Syntax	Gauge32
Description	Total number of nodes (Radio Equipments) currently connected and managed by the RMU. A connected node is defined as a node for which the ORI TCP link is fully established and for which the alignment procedure has completed. Corresponds to the number of node entries in the nodeTable.

Table	nodeTable
SNMP OID	1.3.6.1.4.1.50488.2.50.11
Description	List of node entries. The number of nodes is given by the value of numNodes.

For instance, Figure 46 shows the sequence of notifications that are sent by the RMU to the NMS when a new Radio Equipment is connected to the first Swallow CPRI master port.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (160844) 0:26:48.44
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortLinkUp
 AW2S-RMUv6-MIB::swallowPortRole.0.0 = INTEGER: master(1)
 AW2S-RMUv6-MIB::swallowPortLineSpeed.0.0 = INTEGER: option5(5)

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (160844) 0:26:48.44
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortStatusChanged
 AW2S-RMUv6-MIB::swallowPortStatus.0.0 = INTEGER: operational(4)
 AW2S-RMUv6-MIB::swallowPortStatusInfo.0.0 = STRING: "CPRI link up (option 5: 4915.2Mbps)"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (161464) 0:26:54.64
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeConnected
 AW2S-RMUv6-MIB::nodeProductName.0.0.0 = STRING: PRB000066-A-17360001
 AW2S-RMUv6-MIB::nodeHardwareVer.0.0.0 = STRING: RRH LTE 2x2 Band5 43dBm
 AW2S-RMUv6-MIB::numNodes.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (161464) 0:26:54.64
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeStatusChanged
 AW2S-RMUv6-MIB::nodeStatus.0.0.0 = INTEGER: operational(4)
 AW2S-RMUv6-MIB::nodeStatusInfo.0.0.0 = ""
```

Figure 46: Node connected notifications

12.7.1.1 Node entry

Each connected Radio Equipment can be managed individually; they all have their own set of parameters and monitoring elements.

Table entry	nodeEntry
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1
Indexes	nodeBoardIndex, nodePortIndex, nodeHopIndex
Description	Connected node (radio equipment) entry containing management information applicable to an ORI-managed radio equipment. The values of nodeBoardIndex, nodePortIndex and nodeHopIndex uniquely defines the location of the node in the fronthaul topology.

Read-Only Variable	nodeBoardIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.1.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32
Description	Device 'minor' identifier of the Swallow board to which the node is connected.

Read-Only Variable	nodePortIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.2.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32
Description	Index of the Swallow port to which the node is connected (either directly or indirectly in the daisy-chaining case). This value also corresponds to the physically labeled index of the Swallow CPRI port (e.g. SFP_1 if nodePortIndex is 1).

Read-Only Variable	nodeHopIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.3.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32
Description	Number of hops (networking radio equipments), in CPRI daisy-chaining topology between the Swallow CPRI master port and this node's CPRI slave port. This index identifies the position of the node in the daisy-chain for the Swallow port. If nodeHopIndex is 0, then the node is directly connected to the Swallow port.

Read-Only Variable	nodeStatus
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.4.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	RMUObjectStatus
Description	Node (radio equipment) status.
CLI equiv.	swa:X/port:Y/node:Z info

Read-Only Variable	nodeStatusInfo
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.5.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	InfoString
Description	Node (radio equipment) status information details. May be empty if no relevant information is available.
CLI equiv.	swa:X/port:Y/node:Z info

Notification	nodeStatusChanged
SNMP OID	1.3.6.1.4.1.50488.2.50.14
Variables	nodeStatus, nodeStatusInfo
Description	This notification is emitted when a node's status has changed. The included values of nodeStatus and nodeStatusInfo provide the new status and possibly textual information regarding this new status.

Read-Write Variable	nodeEnable
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.6.nodeBoardIndex(X)

	.nodePortIndex(Y).nodeHopIndex(Z)
Syntax	TruthValue
Description	Sets the enablement of this radio equipment for signal path creation. When set to true(1), the node is enabled for RF operation and all functionalities are available. However, when set to false(2), the node is not configured for RF operation (disabled). When disabled, the node can still accept user commands such as file transfers and reset, but no signal path is created following RAT C&M signal path allocation requests.
CLI equiv.	swa:X/port:Y/node:Z enabled [true|false]

Read-Only Variable	**nodeIpAddr**
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.7.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	IpAddress
Description	IPv4 address of the node's Ethernet over CPRI interface. This is the IP address allocated by the host's DHCP server, and is used for the ORI TCP connection.
CLI equiv.	swa:X/port:Y/node:Z info

Read-Only Variable	**nodeTcpPort**
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.8.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32
Description	TCP port used for the ORI TCP connection. The TCP port is generated by the node and retrieved via DHCP option 43 according to ORI specification.
CLI equiv.	swa:X/port:Y/node:Z info

Some management parameters have special effects on the Radio Equipment as they trigger specific ORI procedures. As these ORI procedures can take some time to complete, the RMU implements message-based SNMP notifications to indicate the completion of a command to the NMS.

Typically, the procedure is started by issuing a write command to an SNMP "Read-Write Command" variable (i.e. nodeReset). Reading back this variable retrieves an unique ORI message identifier "msgUID" associated to the command that was just issued. At a later time, when the procedure completes on the Radio Equipment side, a SNMP notification is emitted by the RMU (i.e. nodeResetDone). This notification includes the below payload data, which helps identifying the source of the message, as well as the procedure result.

Notification Payload	**nodeMsgUID**
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.48.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	DisplayString
Description	ORI procedure response message identifier. This variable is attached to the payload of an ORI procedure response notification, and can be used to identify to which request the notification corresponds.

Notification Payload	nodeMsgSuccess
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.49.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	TruthValue
Description	ORI procedure response result. This variable is attached to the payload of an ORI procedure response notification, and informs if the procedure completed successfully (value is true(1)) or failed (value is false(2)). Note that when it is possible for a procedure to fail partially, in that case value is false(2) and the nodeMsgInfo may provide additional information.

Notification Payload	nodeMsgInfo
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.50.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	InfoString
Description	ORI procedure response information string. This variable is attached to the payload of an ORI procedure response notification, and gives a detailed description of the procedure's effects or failure reason.

12.7.1.2 Product versioning management

Product versioning is an important feature of a Radio Equipment's lifetime management. The following read-only parameters allow the NMS to identify hardware products as well as the software versions of those nodes.
For instance, the following snmpwalk commands may be used to query the product names and active software versions of all connected Radio Equipments:

```
user@shuttle:~$ snmpwalk -v 2c -c public -m ALL localhost nodeIpAddr
AW2S-RMUv6-MIB::nodeIpAddr.0.0.0 = IpAddress: 192.168.127.10
AW2S-RMUv6-MIB::nodeIpAddr.0.1.0 = IpAddress: 192.168.128.12
user@shuttle:~$ snmpwalk -v 2c -c public -m ALL localhost nodeProductName
AW2S-RMUv6-MIB::nodeProductName.0.0.0 = STRING: PRB000066-A-17360001
AW2S-RMUv6-MIB::nodeProductName.0.1.0 = STRING: PRB000068-D-19260001
user@shuttle:~$ snmpwalk -v 2c -c public -m ALL localhost nodeHardwareVer
AW2S-RMUv6-MIB::nodeHardwareVer.0.0.0 = STRING: RRH LTE 2x2 Band5 43dBm
AW2S-RMUv6-MIB::nodeHardwareVer.0.1.0 = STRING: RRH LTE 2x2 Band40 43dBm
user@shuttle:~$ snmpwalk -v 2c -c public -m ALL localhost nodeActiveSwVersion
AW2S-RMUv6-MIB::nodeActiveSwVersion.0.0.0 = STRING: cadc_032
AW2S-RMUv6-MIB::nodeActiveSwVersion.0.1.0 = STRING: cadc_032
user@shuttle:~$
```

Figure 47: Reading the product version

Read-Only Variable	nodeVendor
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.9.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	DisplayString

Description	ORI vendor code for this node, usually 3 letters, retrieved via DHCP option 43 according to ORI specification.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodeProductName**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.10.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Node product name that uniquely defines the product. The string is a concatenation of the product part number, revision and serial number. Those versioning elements are obtained during initial node alignment via the ORI version query procedure.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodeHardwareVer**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.11.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Node hardware version (variant) descriptive string. This versioning element is obtained during initial node alignment via the ORI version query procedure.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodeActiveSwVersion**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.12.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Active software package image version. This is the software release currently running on the node. This versioning element is obtained during initial node alignment via the ORI version query procedure.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodePassiveSwVersion**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.13.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Passive software package image version. This software release is available on the node but not currently running (it can be set to active by using the `nodeSwActivate` command). This versioning element is obtained during initial node alignment via the ORI version query procedure.
CLI equiv.	`swa:X/port:Y/node:Z info`

The RMU offers the possibility to reset (software reboot) a remote Radio Equipment through the ORI reset procedure, by using the `nodeReset` command, as shown in Figure 48.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost nodeReset.0.0.0 s RST
Unexpected index type: 7 nodeBoardIndex 0.0.0
AW2S-RMUv6-MIB::nodeReset.0.0.0 = STRING: RST
user@shuttle:~$
```

Figure 48: Node reset command

Following this command, the Radio Equipment is disconnected from the RMU and begins a software reboot. The resulting sequence of notifications is displayed in Figure 49.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (467819) 1:17:58.19
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeResetDone
 AW2S-RMUv6-MIB::nodeMsgUID.0.0.0 = STRING: 11
 AW2S-RMUv6-MIB::nodeMsgSuccess.0.0.0 = INTEGER: true(1)
 AW2S-RMUv6-MIB::nodeMsgInfo.0.0.0 = STRING: "ORI reset (msgUID=11) success: deleting node"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (467819) 1:17:58.19
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeDisconnected
 AW2S-RMUv6-MIB::nodeProductName.0.0.0 = STRING: PRB000066-A-17360001
 AW2S-RMUv6-MIB::nodeHardwareVer.0.0.0 = STRING: RRH LTE 2x2 Band5 43dBm
 AW2S-RMUv6-MIB::numNodes.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (468944) 1:18:09.44
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortLinkDown
 AW2S-RMUv6-MIB::swallowPortRole.0.0 = INTEGER: master(1)
 AW2S-RMUv6-MIB::swallowPortLineSpeed.0.0 = INTEGER: option5(5)

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (468944) 1:18:09.44
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortStatusChanged
 AW2S-RMUv6-MIB::swallowPortStatus.0.0 = INTEGER: preOperational(3)
 AW2S-RMUv6-MIB::swallowPortStatusInfo.0.0 = STRING: "CPRI link down"
```

Figure 49: Node reset sequence of notifications

Read-Write Command	**nodeReset**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.14.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Writing the value `RST` on this field will trigger the ORI Radio Equipment Reset procedure, causing the Radio Equipment to reboot. Writing any other value results in a 'Bad Value' error. After a successful write, a GET command on this field returns the ORI message identifier (`nodeMsgUID`) of the reset request sent to the Radio Equipment. The `nodeResetDone` notification (which also includes the `nodeMsgUID` value) will then be emitted by the RMU when the Radio Equipment has acknowledged the request, before initiating a software reboot.

	In case of 'Bad Value' failure of the SET command, a GET command will return an information string explaining the failure reason.
CLI equiv.	`swa:X/port:Y/node:Z reset`

Notification	**nodeResetDone**
SNMP OID	`1.3.6.1.4.1.50488.2.50.15`
Variables	`nodeMsgUID, nodeMsgSuccess, nodeMsgInfo`
Description	This notification is emitted when the ORI Radio Equipment Reset procedure (following a `nodeReset` command) has completed. The variable binding `nodeMsgUID` can help matching this notification to the request command. Variable bindings `nodeMsgSuccess` and `nodeMsgInfo` inform of the procedure result with more details. If `nodeMsgSuccess` is `true(1)`, then the Radio Equipment initiates a software reboot.

Through the usage of SNMP commands `nodeSwUpgrade` and `nodeSwActivate`, the NMS can update the software loaded into a Radio Equipment.

For instance, the commands displayed in Figure 50 shows the software upgrade procedure.
In this example, we suppose that the valid Radio Equipment software package `release_cadc_032.zip` is available in the `releases` directory of the FTP server hosted locally. A FTP link pointing to this FTP directory is setup in the RMU. After the `nodeSwUpgrade` command is successfully sent, reading back this parameter gives a procedure message identifier `msgUID` value of `52`. This means that the software upgrade procedure has started on the Radio Equipment side: the remote equipment is in the process of downloading the package file from the FTP then stores it into non-volatile memory.

```
user@shuttle:~$ snmptable -v 2c -c public -m ALL -Cb localhost ftpLinkTable
SNMP table: AW2S-RMUv6-MIB::ftpLinkTable

 Index   ObjectName      Alias      Host UserName   Password  Directory
     0 "ftp/link:0" "Releases" 127.0.0.1   "user" "default0" "releases"
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost nodeSwUpgrade.0.0.0 s "ftp/link:0 release_cadc_032.zip"
Unexpected index type: 7 nodeBoardIndex 0.0.0
AW2S-RMUv6-MIB::nodeSwUpgrade.0.0.0 = STRING: "ftp/link:0 release_cadc_032.zip"
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost nodeSwUpgrade.0.0.0
Unexpected index type: 7 nodeBoardIndex 0.0.0
AW2S-RMUv6-MIB::nodeSwUpgrade.0.0.0 = STRING: "52"
user@shuttle:~$
```

Figure 50: Node software upgrade procedure

Once the Radio Equipment has finished downloading and storing the software package, the `nodeSwUpgradeDone` notification is emitted, as shown in Figure 51. The `nodeMsgUID` payload variable allows us to link this notification to the previous `nodeSwUpgrade` command.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (562257) 1:33:42.57
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeSwUpgradeDone
 AW2S-RMUv6-MIB::nodeMsgUID.0.0.0 = STRING: 52
 AW2S-RMUv6-MIB::nodeMsgSuccess.0.0.0 = INTEGER: true(1)
 AW2S-RMUv6-MIB::nodeMsgInfo.0.0.0 = STRING: "ORI software upgrade download (msgUID=52) successfully complet
ed"
```

Figure 51: Node software upgrade completion

At this point, the newly downloaded software package is stored as the passive image on the Radio Equipment. As per ORI specification, this new version needs to be activated for usage by the Radio Equipment. This can be done by issuing the `nodeSwActivate` command, as seen in Figure 52.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost nodeSwActivate.0.0.0 s ACT
Unexpected index type: 7 nodeBoardIndex 0.0.0
AW2S-RMUv6-MIB::nodeSwActivate.0.0.0 = STRING: ACT
user@shuttle:~$
```

Figure 52: Node software activation procedure

On activation completion, the Radio Equipment sets its current passive image as the new active image, and the previous active image becomes the passive image. The Radio Equipment is then disconnected from the RMU and initiates a software reboot to complete the procedure. The resulting sequence of notifications is displayed in Figure 53.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (574638) 1:35:46.38
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeSwActivateDone
 AW2S-RMUv6-MIB::nodeMsgUID.0.0.0 = STRING: 58
 AW2S-RMUv6-MIB::nodeMsgSuccess.0.0.0 = INTEGER: true(1)
 AW2S-RMUv6-MIB::nodeMsgInfo.0.0.0 = STRING: "ORI software activation (msgUID=58) success: deleting node"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (574638) 1:35:46.38
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeDisconnected
 AW2S-RMUv6-MIB::nodeProductName.0.0.0 = STRING: PRB000066-A-17360001
 AW2S-RMUv6-MIB::nodeHardwareVer.0.0.0 = STRING: RRH LTE 2x2 Band5 43dBm
 AW2S-RMUv6-MIB::numNodes.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (575843) 1:35:58.43
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortLinkDown
 AW2S-RMUv6-MIB::swallowPortRole.0.0 = INTEGER: master(1)
 AW2S-RMUv6-MIB::swallowPortLineSpeed.0.0 = INTEGER: option5(5)

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (575843) 1:35:58.43
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::swallowPortStatusChanged
 AW2S-RMUv6-MIB::swallowPortStatus.0.0 = INTEGER: preOperational(3)
 AW2S-RMUv6-MIB::swallowPortStatusInfo.0.0 = STRING: "CPRI link down"
```

Figure 53: Node software activation completion

Read-Write Command	**nodeSwUpgrade**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.15.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`SingleFileTransferString`
Description	Issuing a SET command on this field will trigger the ORI Software Upgrade Preparation procedure, causing the Radio Equipment to download and store a new software package into the non-volatile memory passive image. The value for the SET command contains a reference to the FTP link that the Radio Equipment will use to download the new software package, as well as the actual software package file name, e.g. `ftp/link:1 release.zip`. After a successful write, a GET command on this field returns the ORI message identifier (`nodeMsgUID`) of the software upgrade request sent to the Radio Equipment. The `nodeSwUpgradeDone` notification (which also includes the `nodeMsgUID` value) will be

	emitted by the RMU when the Radio Equipment has completed the software upgrade procedure, either successfully or not. In case of 'Bad Value' failure of the SET command, a GET command will return an information string explaining the failure reason. Note that after a successful software upgrade procedure, the new software image needs to be activated (using `nodeSwActivate` command) for the Radio Equipment to finalize and start using the new software image.
CLI equiv.	`swa:X/port:Y/node:Z sw-upgrade <ftpLinkPathName|ftpLinkAlias> <swPkgName>`

Notification	**nodeSwUpgradeDone**
SNMP OID	`1.3.6.1.4.1.50488.2.50.16`
Variables	`nodeMsgUID, nodeMsgSuccess, nodeMsgInfo`
Description	This notification is emitted when the ORI Software Upgrade Preparation procedure (following a `nodeSwUpgrade` command) has completed. The variable binding `nodeMsgUID` can help matching this notification to the request command. Variable bindings `nodeMsgSuccess` and `nodeMsgInfo` inform of the procedure result with more details. If `nodeMsgSuccess` is `true(1)`, then the Radio Equipment has stored the downloaded software image into non-volatile memory as a passive image.

Read-Write Command	**nodeSwActivate**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.16.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`DisplayString`
Description	Writing the value `ACT` on this field will trigger the ORI Software Activation procedure, causing the Radio Equipment to activate the passive software image. Writing any other value results in a 'Bad Value' error. After a successful write, a GET command on this field returns the ORI message identifier (`nodeMsgUID`) of the reset request sent to the Radio Equipment. The `nodeSwActivateDone` notification (which also includes the `nodeMsgUID` value) will be emitted by the RMU when the Radio Equipment has activated the passive software image, before initiating a software reboot. In case of 'Bad Value' failure of the SET command, a GET command will return an information string explaining the failure reason.
CLI equiv.	`swa:X/port:Y/node:Z sw-activate`

Notification	**nodeSwActivateDone**
SNMP OID	`1.3.6.1.4.1.50488.2.50.17`
Variables	`nodeMsgUID, nodeMsgSuccess, nodeMsgInfo`
Description	This notification is emitted when the ORI Software Activation procedure (following a `nodeSwActivate` command) has completed. The variable binding `nodeMsgUID` can help matching this notification to the request command.

	Variable bindings `nodeMsgSuccess` and `nodeMsgInfo` inform of the procedure result with more details. If `nodeMsgSuccess` is `true(1)`, then the Radio Equipment has activated the passive software image and initiates a software reboot.

12.7.1.3 Health monitoring

Health monitoring of a Radio Equipment is the simple task of regularly checking some parameters that may affect the functionality of the equipment. For instance, a common variable is the product's temperature.

```
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost nodeUptime.0.1.0 nodeTemperature.0.1.0 nodeInputVoltage.0.1.0 nodePowerUsage.0.1.0
Unexpected index type: 7 nodeBoardIndex 0.1.0
Unexpected index type: 7 nodeBoardIndex 0.1.0
Unexpected index type: 7 nodeBoardIndex 0.1.0
Unexpected index type: 7 nodeBoardIndex 0.1.0
AW2S-RMUv6-MIB::nodeUptime.0.1.0 = Timeticks: (623800) 1:43:58.00
AW2S-RMUv6-MIB::nodeTemperature.0.1.0 = INTEGER: 39 degC
AW2S-RMUv6-MIB::nodeInputVoltage.0.1.0 = INTEGER: 48180 mVolts
AW2S-RMUv6-MIB::nodePowerUsage.0.1.0 = INTEGER: 40 Watts
user@shuttle:~$
```

Figure 54: Node health monitors

Read-Only Variable	**nodeUptime**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.17.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`TimeTicks`
Description	"The time (in hundredths of a second) since the node was powered on and finished its boot sequence. This parameter is updated every 30 seconds.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodeTemperature**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.18.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`Integer32` (degC)
Description	Product temperature of the node. This parameter is updated every 30 seconds.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	**nodeInputVoltage**
SNMP OID	`1.3.6.1.4.1.50488.2.50.11.1.19.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)`
Syntax	`Integer32` (mVolts)
Description	Node power-supply input voltage, measured at the product power connector. This parameter is updated every 30 seconds.
CLI equiv.	`swa:X/port:Y/node:Z info`

Read-Only Variable	nodePowerUsage
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.20.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Integer32 (Watts)
Description	Node (instantaneous) total power dissipation, calculated as the product of the nodeInputVoltage and measured current at the product's power connector. This parameter is updated every 30 seconds.
CLI equiv.	swa:X/port:Y/node:Z info

All the following read-only variables correspond to parameters and functional states of ORI objects that reside inside the Radio Equipment. They may prove useful for debugging issues that may appear at the Radio Equipment's side.

Read-Only Variable	nodeCpriSlaveLabel
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.21.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	DisplayString
Description	Name of the physical CPRI slave SFP connector as indicated on the radio equipment housing.
CLI equiv.	swa:X/port:Y/node:Z info

Read-Only Variable	nodeCpriSlaveFST
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.22.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	OriFST
Description	ORI-defined functional state of the CPRI slave interface. This parameter is updated asynchronously via indications coming in from the radio equipment.

Read-Only Variable	nodeCpriSlaveToffset
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.23.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32 (ns)
Description	Value of the CPRI slave port's Toffset timing parameter, used for calculation of the cable delay. This parameter is defined in the CPRI specification section Link Delay Accuracy and Cable Delay Calibration. This parameter is updated every 30 seconds.

Read-Only Variable	nodeCpriSlaveSfpTxPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.24.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Integer32 (dBm/100)
Description	Value of the optical power transmitted by the CPRI slave SFP module.

	This parameter is updated every 30 seconds.

Read-Only Variable	nodeCpriSlaveSfpRxPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.25.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Integer32 (dBm/100)
Description	Value of the optical power received by the CPRI slave SFP module. This parameter is updated every 30 seconds.

Read-Only Variable	nodeCpriMasterLabel
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.26.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	DisplayString
Description	Name of the physical CPRI master SFP connector as indicated on the radio equipment housing. If no CPRI master is available on the node, then the string is empty.
CLI equiv.	swa:X/port:Y/node:Z info

Read-Only Variable	nodeCpriMasterFST
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.27.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	OriFST
Description	ORI-defined functional state of the CPRI master interface. This parameter is updated asynchronously via indications coming in from the radio equipment. If no CPRI master is available on the node, then the value is disabled(5).

Read-Only Variable	nodeCpriMasterT14
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.28.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32 (ns)
Description	Value of the CPRI master port's T14 timing parameter, used for calculation of the cable delay. This parameter is defined in the CPRI specification section Link Delay Accuracy and Cable Delay Calibration. This parameter is updated every 30 seconds. If no CPRI master is available on the node, then the value is invalid.

Read-Only Variable	nodeCpriMasterSfpTxPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.29.nodeBoardIndex(X).nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Integer32 (dBm/100)
Description	Value of the optical power transmitted by the CPRI master SFP module. This parameter is updated every 30 seconds. If no CPRI master is available on the node, then the value is invalid.

Read-Only Variable	nodeCpriMasterSfpRxPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.30.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Integer32 (dBm/100)
Description	Value of the optical power received by the CPRI master SFP module. This parameter is updated every 30 seconds. If no CPRI master is available on the node, then the value is invalid.

Read-Only Variable	nodeNumAntenna
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.31.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32
Description	Number of antenna ports populated by this radio equipment.

Read-Only Variable	nodeDlRouteFST
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.35.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	OriFST
Description	ORI-defined functional state of the I/Q downlink route between the CPRI slave port and the CPRI master port, when the radio equipment is also working as a networking RE (daisy-chaining configuration). If no downlink route is available on the node, then the value is disabled(5).

Read-Only Variable	nodeDlRouteLatency
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.36.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32 (ns)
Description	Latency of the digital I/Q data through the I/Q downlink route. This is the time it takes for I/Q data to be routed from the CPRI slave port SFP connector Rx to the CPRI master SFP connector Tx. If no downlink route is available on the node, then the value is invalid.

Read-Only Variable	nodeUlRouteFST
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.37.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	OriFST
Description	ORI-defined functional state of the I/Q uplink route between the CPRI slave port and the CPRI master port, when the radio equipment is also working as a networking RE (daisy-chaining configuration). If no uplink route is available on the node, then the value is disabled(5).

Read-Only Variable	nodeUlRouteLatency
SNMP OID	1.3.6.1.4.1.50488.2.50.11.1.38.nodeBoardIndex(X) .nodePortIndex(Y).nodeHopIndex(Z)
Syntax	Gauge32 (ns)
Description	Latency of the digital I/Q data through the I/Q uplink route. This is the time it takes for I/Q data to be routed from the CPRI master port SFP connector Rx to the CPRI slave SFP connector Tx. If no uplink route is available on the node, then the value is invalid.

12.7.2 RF antenna ports

The antennaTable table references all RF antenna ports currently managed by the RMU. Those RF antenna ports correspond to the physical RF connectors on the different Radio Equipments. The list of RF antenna ports is thus updated when Radio Equipment are connected or disconnected from the RMU. For instance, Figure 55 below shows a snmptable view of the RF antenna ports.

```
user@shuttle:~$ snmptable -v 2c -c public -m ALL -Cb localhost antennaTable
SNMP table: AW2S-RMUv6-MIB::antennaTable

 BoardIndex PortIndex HopIndex Index          ProductName Label Enable    OutputPwr      InputPwr      ReturnLoss NumTxSigPaths NumRxSigPaths
          0         0        0     0 PR8000066-A-17360001  ANT1   true 1730 dBm/100 -9560 dBm/100 -327680 dB/100            1            1
          0         0        0     1 PR8000066-A-17360001  ANT2   true 1760 dBm/100 -9570 dBm/100 -327680 dB/100            1            1
          0         1        0     0 PR8000068-D-19260001  ANT1   true -670 dBm/100 -9180 dBm/100 -327680 dB/100            0            0
          0         1        0     1 PR8000068-D-19260001  ANT2   true -480 dBm/100 -9140 dBm/100 -327680 dB/100            0            0
user@shuttle:~$ _
```

Figure 55: RF antenna ports table display

Read-Only Variable	numAntennas
SNMP OID	1.3.6.1.4.1.50488.2.50.20.0
Syntax	Gauge32
Description	Total number of antenna ports available for signal paths mapping in the RMU. Each Radio Equipment can populate multiple antenna ports.

Table	antennaTable
SNMP OID	1.3.6.1.4.1.50488.2.50.21
Description	List of antenna port entries. The number of antenna ports is given by the value of numAntennas.

12.7.2.1 RF antenna port entry

Each RF antenna port can be managed individually; they all have their own set of parameters and monitoring elements.

Table entry	antennaEntry
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1
Indexes	antennaBoardIndex, antennaPortIndex, antennaHopIndex, antennaIndex
Description	Antenna port entry containing management information applicable to a Radio Equipment antenna port. The values of antennaBoardIndex, antennaPortIndex and antennaHopIndex uniquely defines the location of the parent node in the fronthaul topology. The antennaIndex value further determines the antenna port identifier in the Radio Equipment.

Read-Only Variable	antennaBoardIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.1.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Device 'minor' identifier of the Swallow board to which the parent node is connected.

Read-Only Variable	antennaPortIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.2.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Index of the Swallow port to which the parent node is connected (either directly or indirectly in the daisy-chaining case). This value also corresponds to the physically labeled index of the Swallow CPRI port (e.g. SFP_1 if antennaPortIndex is 1).

Read-Only Variable	antennaHopIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.3.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Number of hops (networking radio equipments), in CPRI daisy-chaining topology between the Swallow CPRI master port and the parent node's CPRI slave port. This index identifies the position of the parent node in the daisy-chain for the Swallow port. If antennaHopIndex is 0, then the parent node is directly connected to the Swallow port.

Read-Only Variable	antennaIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.4.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Unique value for each antenna port of a given Radio Equipment, used to index the list of antenna port entries.

Read-Only Variable	antennaProductName
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.5.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	DisplayString
Description	Product name of the parent node. The string is a concatenation of the product part number, revision and serial number.

Read-Only Variable	antennaLabel
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.6.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	DisplayString
Description	Name of the physical antenna port as indicated on the radio equipment housing.

On NMS request, a RF antenna port can be enabled or disabled independently from each other, on any specific Radio Equipment.

Read-Write Variable	antennaEnable
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.7.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	TruthValue
Description	Sets the enablement of this antenna port. When set to true(1), the antenna port is enabled for operation, and accepts mapping of RF signal paths. When set to false(2), the antenna port is disabled for operation, and refuses mapping of RF signal paths.
CLI equiv.	swa:X/port:Y/node:Z/ant:U enabled [true|false]

Read-Only Variable	antennaOutputPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.8.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Integer32 (dBm/100)
Description	Measured output power at the radio equipment antenna port. The power measurement is done by the radio equipment as the integrated full-bandwidth transmitted power within a 10 milliseconds window. This parameter is updated every 30 seconds.
CLI equiv.	swa:X/port:Y/node:Z/ant:U info

Read-Only Variable	antennaInputPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.9.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Integer32 (dBm/100)
Description	Measured input power at the radio equipment antenna port. The power measurement is done by the radio equipment as the integrated full-bandwidth received power within a 10 milliseconds window. This parameter is updated every 30 seconds.
CLI equiv.	swa:X/port:Y/node:Z/ant:U info

Read-Only Variable	antennaReturnLoss
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.10.antennaBoardIndex(X) .antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Integer32 (dB/100)
Description	Return loss measured at the radio equipment antenna port. The return loss is measured by the radio equipment as a slow exponential moving average of the full-bandwidth transmitted power divided by the reverse power. This parameter is updated every 30 seconds.
CLI equiv.	swa:X/port:Y/node:Z/ant:U info

Read-Only Variable	antennaNumTxSigPaths
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.11.antennaBoardIndex(X).antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Number of Tx signal paths mapped and allocated on this antenna port.
CLI equiv.	swa:X/port:Y/node:Z/ant:U info -v

Read-Only Variable	antennaNumRxSigPaths
SNMP OID	1.3.6.1.4.1.50488.2.50.21.1.12.antennaBoardIndex(X).antennaPortIndex(Y).antennaHopIndex(Z).antennaIndex(U)
Syntax	Gauge32
Description	Number of Rx signal paths mapped and allocated on this antenna port.
CLI equiv.	swa:X/port:Y/node:Z/ant:U info -v

12.7.3 RF signal paths

RF signal paths are signal paths allocated on Radio Equipment based on RAT C&M requests currently pending in the RMU.

As long as the Radio Equipment is connected and enabled, and that the target RF antenna port is also enabled, a signal path is created on the Radio Equipment for the relevant RAT C&M request. A signal path transports the I/Q data between the CPRI port and the antenna port, with appropriate signal processing.

The sigPathTable list all the currently allocated RF signal paths.

The sigPathCreated notification is emitted when a signal path is created (based on a RAT C&M request), on a specific Radio Equipment antenna port, as shown in Figure 56. The variable bindings of the notification provide the signal path linkage.

Figure 56: RF signal path created notification

Notification	sigPathCreated
SNMP OID	1.3.6.1.4.1.50488.2.50.32
Variables	sigPathProductName, sigPathAntennaLabel, sigPathCmDescr, numSigPaths
Description	The sigPathCreated notification is emitted when a new signal path has been allocated on a Radio Equipment (a new signal path entry has been added to the sigPathTable). The sigPathProductName and sigPathAntennaLabel values identify on which product and antenna port the signal path has been created. Variable binding sigPathCmDescr gives a short description of the signal path. The numSigPaths is also updated accordingly.

Notification	sigPathDeleted
SNMP OID	1.3.6.1.4.1.50488.2.50.33
Variables	sigPathProductName, sigPathAntennaLabel, sigPathCmDescr, numSigPaths
Description	The sigPathDeleted notification is emitted when a signal path has been removed from a Radio Equipment (a signal path entry has been removed from the sigPathTable). The sigPathProductName and sigPathAntennaLabel values identify from which product and antenna port the signal path has been removed. Variable binding sigPathCmDescr gives a short description of the signal path. The numSigPaths is also updated accordingly.

Read-Only Variable	numSigPaths
SNMP OID	1.3.6.1.4.1.50488.2.50.30.0
Syntax	Gauge32
Description	Total number of RF signal paths managed by the RMU.

Table	sigPathTable
SNMP OID	1.3.6.1.4.1.50488.2.50.31
Description	List of RF signal path entries. The number of RF signal paths is given by the value of numSigPaths.

12.7.3.1 RF signal path entry

Each RF signal path can be managed individually; they all have their own set of parameters and monitoring elements.

Table entry	sigPathEntry
SNMP OID	1.3.6.1.4.1.50488.2.50.31.1
Indexes	sigPathBoardIndex, sigPathPortIndex, sigPathHopIndex, sigPathAntennaIndex, sigPathDirection, sigPathIndex
Description	Radio Equipment signal path entry containing management information applicable to a RF signal path allocated on a radio equipment managed by the RMU. The values of sigPathBoardIndex, sigPathPortIndex and sigPathHopIndex uniquely defines the location of the parent node in the fronthaul topology.

	The `sigPathAntennaIndex` and `sigPathDirection` values give the antenna port and direction (TX or RX) of the signal path.

Read-Only Variable	**sigPathBoardIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.1.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`Gauge32`
Description	Device '`minor`' identifier of the Swallow board to which the parent node is connected.

Read-Only Variable	**sigPathPortIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.2.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`Gauge32`
Description	Index of the Swallow port to which the parent node is connected (either directly or indirectly in the daisy-chaining case). This value also corresponds to the physically labeled index of the Swallow CPRI port (e.g. SFP_1 if `sigPathPortIndex` is 1).

Read-Only Variable	**sigPathHopIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.3.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`Gauge32`
Description	Number of hops (networking radio equipments), in CPRI daisy-chaining topology between the Swallow CPRI master port and the parent node's CPRI slave port. This index identifies the position of the parent node in the daisy-chain for the Swallow port. If `sigPathHopIndex` is 0, then the parent node is directly connected to the Swallow port.

Read-Only Variable	**sigPathAntennaIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.4.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`Gauge32`
Description	Index of the parent antenna port on which this signal path is allocated.

Read-Only Variable	**sigPathDirection**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.5.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`INTEGER` (enumerated)
Description	Indicates the direction of the signal path, relative to the radio equipment.

	The value `tx(1)` means the signal is going to the RF port (digital -> analog), while the value `rx(2)` means the signal is coming in from the RF port (analog -> digital).

Read-Only Variable	**sigPathIndex**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.6.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`Gauge32`
Description	Unique value for each signal path for a given antenna port, used to index the list of signal path entries.

Read-Only Variable	**sigPathStatus**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.7.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`RMUObjectStatus`
Description	Signal path status.
CLI equiv.	`swa:X/port:Y/node:Z/ant:U/tx:V info` `swa:X/port:Y/node:Z/ant:U/rx:V info`

Read-Only Variable	**sigPathStatusInfo**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.8.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`InfoString`
Description	Signal path status information details. May be empty if no relevant information is available.
CLI equiv.	`swa:X/port:Y/node:Z/ant:U/tx:V info` `swa:X/port:Y/node:Z/ant:U/rx:V info`

Notification	**sigPathStatusChanged**
SNMP OID	`1.3.6.1.4.1.50488.2.50.34`
Variables	`sigPathStatus, sigPathStatusInfo`
Description	This notification is emitted when a signal path's status has changed. The included values of `sigPathStatus` and `sigPathStatusInfo` provide the new status and possibly textual information regarding this new status.

12.7.3.2 RF signal path linkage

The signal path linkage is a set of static read-only parameters that can be used to identify the location and the type of the signal path. It provides identification of the parent Radio Equipment (on which node the signal path is allocated), as well as the RAT C&M signal path request it corresponds to. For instance, Figure 57 below is an extract of `snmptable` for the `sigPathTable`, that shows these parameters.

```
ProductName AntennaLabel          CmName                                                  CmDescr
PRB000066-A-17360001 ANT1 "cm/service:0/tx:0" TX LTE 20MHz FDD @ arfcn=2525 / 30.00dBm
PRB000066-A-17360001 ANT1 "cm/service:0/rx:0"          RX LTE 20MHz FDD @ arfcn=20525
PRB000066-A-17360001 ANT2 "cm/service:0/tx:1" TX LTE 20MHz FDD @ arfcn=2525 / 30.00dBm
PRB000066-A-17360001 ANT2 "cm/service:0/rx:1"          RX LTE 20MHz FDD @ arfcn=20525
```

Figure 57: RF signal paths linkage table extract

Read-Only Variable	**sigPathProductName**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.9.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`DisplayString`
Description	Product name of the parent node. The string is a concatenation of the product part number, revision and serial number.

Read-Only Variable	**sigPathAntennaLabel**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.10.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`DisplayString`
Description	Name of the physical antenna port on which this signal path is allocated, as indicated on the radio equipment housing.

Read-Only Variable	**sigPathCmName**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.11.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)`
Syntax	`AliasString`
Description	The object name of the RAT C&M request for which this signal path is allocated. The RAT C&M object name is in the form '`cm/service:X/Y:Z`' where `X` is the identifier of the RAT C&M service, `Y` can be either '`tx`' or '`rx`' depending on `sigPathDirection` and `Z` is the `cmPathIndex` value. I.e. '`cm/service:1/tx:2`'.
CLI equiv.	`swa:X/port:Y/node:Z/ant:U/tx:V info` `swa:X/port:Y/node:Z/ant:U/rx:V info`

Read-Only Variable	**sigPathCmDescr**
SNMP OID	`1.3.6.1.4.1.50488.2.50.31.1.12.sigPathBoardIndex(X)`

	.sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U) .sigPathDirection.sigPathIndex(V)
Syntax	DisplayString
Description	A string describing the signal path, as defined by the RAT C&M signal path request.
CLI equiv.	swa:X/port:Y/node:Z/ant:U/tx:V info swa:X/port:Y/node:Z/ant:U/rx:V info

12.7.3.3 RF signal path operation

The following variables manage the status of the ORI object corresponding to this RF signal path. For instance, Figure 58 below is an extract of `snmptable` for the `sigPathTable`, that shows these parameters.

```
ProcessStep  Latency     MeasuredPwr          FST
Operational 26437 ns    1739 dBm/100 operational
Operational  3326 ns   -9990 dBm/100 operational
Operational 26439 ns    1710 dBm/100 operational
Operational  3314 ns  -10010 dBm/100 operational
```

Figure 58: RF signal paths operation table extract

Read-Only Variable	**sigPathProcessStep**
SNMP OID	1.3.6.1.4.1.50488.2.50.31.1.13.sigPathBoardIndex(X) .sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U) .sigPathDirection.sigPathIndex(V)
Syntax	DisplayString
Description	String describing the current ORI RF signal path object management state machine step.
CLI equiv.	swa:X/port:Y/node:Z/ant:U/tx:V info swa:X/port:Y/node:Z/ant:U/rx:V info

Read-Only Variable	**sigPathLatency**
SNMP OID	1.3.6.1.4.1.50488.2.50.31.1.14.sigPathBoardIndex(X) .sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U) .sigPathDirection.sigPathIndex(V)
Syntax	Gauge32 (ns)
Description	Latency of the I/Q data for this signal path. For a TX signal path, this is the latency measured from the Radio Equipment CPRI input to antenna port, also named the T2A timing parameter. For a RX signal path, this is the latency measured from the Radio Equipment antenna port to CPRI output, also named the TA3 timing parameter. This parameter is updated every 30 seconds. If the signal path object is not available on the node, then the value is 0.
CLI equiv.	swa:X/port:Y/node:Z/ant:U/tx:V info swa:X/port:Y/node:Z/ant:U/rx:V info

Read-Only Variable	sigPathMeasuredPwr
SNMP OID	1.3.6.1.4.1.50488.2.50.31.1.15.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)
Syntax	Integer32 (dBm/100)
Description	Measured power for this signal path. The power measurement is done by the radio equipment as the integrated whole-band power within a 10 milliseconds window. This parameter is updated every 30 seconds. If the signal path object is not available on the node, then the value is invalid.
CLI equiv.	swa:X/port:Y/node:Z/ant:U/tx:V info swa:X/port:Y/node:Z/ant:U/rx:V info

Read-Only Variable	sigPathFST
SNMP OID	1.3.6.1.4.1.50488.2.50.31.1.16.sigPathBoardIndex(X).sigPathPortIndex(Y).sigPathHopIndex(Z).sigPathAntennaIndex(U).sigPathDirection.sigPathIndex(V)
Syntax	TruthValue
Description	ORI-defined functional state of the node signal path object. This parameter is updated asynchronously via indications coming in from the radio equipment. If the signal path object is not available on the node, then the value is disabled(5).
CLI equiv.	swa:X/port:Y/node:Z/ant:U/tx:V info -v swa:X/port:Y/node:Z/ant:U/rx:V info -v

12.7.4 Fault monitoring

Radio Equipment faults are monitored on the RMU. When a fault becomes active on a remote node, the `faultActivated` notification is triggered.
It may be worth to note that an active fault on a Radio Equipment may or may not (depending on fault type and severity) cause the parent Radio Equipment's `nodeStatus` variable to enter the `errored(2)` state, which in turn may set `rmuMasterAlarm` to `true(1)` and trigger the `rmuMasterAlarmSet` notification.
Figure 59 shows the notifications emitted by the RMU when a Radio Equipment fault causes the RMU master alarm to be set.

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (713828) 1:58:58.28
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::faultActivated
 AW2S-RMUv6-MIB::faultType.0.0.0.extSupplyUndervolt = INTEGER: extSupplyUndervolt(0)
 AW2S-RMUv6-MIB::faultProductName.0.0.0.extSupplyUndervolt = STRING: PRB000066-A-17360001
 AW2S-RMUv6-MIB::faultSeverity.0.0.0.extSupplyUndervolt = INTEGER: failed(2)
 AW2S-RMUv6-MIB::faultDescr.0.0.0.extSupplyUndervolt = STRING: 17.619V
 AW2S-RMUv6-MIB::numFaults.0 = Gauge32: 1

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (713828) 1:58:58.28
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeStatusChanged
 AW2S-RMUv6-MIB::nodeStatus.0.0.0 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::nodeStatusInfo.0.0.0 = STRING: "Fault: External power supply under voltage Failed (17.619V)"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (713828) 1:58:58.28
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::faultActivated
 AW2S-RMUv6-MIB::faultType.0.0.0.txGainFail = INTEGER: txGainFail(4)
 AW2S-RMUv6-MIB::faultProductName.0.0.0.txGainFail = STRING: PRB000066-A-17360001
 AW2S-RMUv6-MIB::faultSeverity.0.0.0.txGainFail = INTEGER: failed(2)
 AW2S-RMUv6-MIB::faultDescr.0.0.0.txGainFail = STRING: Amplifier protected due to PSU input voltage out of range (17.62V)
 AW2S-RMUv6-MIB::numFaults.0 = Gauge32: 2

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (713828) 1:58:58.28
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::nodeStatusChanged
 AW2S-RMUv6-MIB::nodeStatus.0.0.0 = INTEGER: errored(2)
 AW2S-RMUv6-MIB::nodeStatusInfo.0.0.0 = STRING: "Fault: External power supply under voltage Failed (17.619V) / Fault: TX gain
 failure Failed (Amplifier protected due to PSU input voltage out of range (17.62V))"

DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (713844) 1:58:58.44
 SNMPv2-MIB::snmpTrapOID.0 = OID: AW2S-RMUv6-MIB::rmuMasterAlarmSet
 AW2S-RMUv6-MIB::rmuMasterAlarm.0 = INTEGER: true(1)
```

Figure 59: Radio Equipment fault activated

The list of currently active faults is stored in the `faultTable`. When the table is empty, this means that there is no fault active on any Radio Equipment managed by the RMU.

```
user@shuttle:~$ snmptable -v 2c -c public -m ALL -Cb localhost faultTable
SNMP table: AW2S-RMUv6-MIB::faultTable

 BoardIndex PortIndex HopIndex             Type          ProductName Severity                                                  Descr     Duration
          0         0        0 extSupplyUndervolt PRB000066-A-17360001   failed                                          17.619V 0:0:01:25.47
          0         0        0        txGainFail PRB000066-A-17360001   failed Amplifier protected due to PSU input voltage out of range (17.62V) 0:0:01:25.47
user@shuttle:~$
```

Figure 60: Example fault table with active faults

Notification	faultActivated
SNMP OID	1.3.6.1.4.1.50488.2.50.42
Variables	faultType, faultProductName, faultSeverity, faultDescr, numFaults
Description	The faultActivated notification is emitted when a fault enters the active state, or the information contained in an already active fault entry has been updated. The variable bindings faultType and faultProductName informs the type of fault that is now active and on which radio equipment it occurred. The faultSeverity and faultDescr

	details the severity of the fault as well as a descriptive string. The numFaults is also updated accordingly.

Notification	faultCleared
SNMP OID	1.3.6.1.4.1.50488.2.50.43
Variables	faultType, faultProductName, faultSeverity, faultDescr, numFaults
Description	The faultCleared notification is emitted when a fault is cleared (exits the active state). The variable bindings faultType and faultProductName informs the type of fault that was cleared and on which radio equipment it occurred. The faultSeverity and faultDescr details the severity of the fault before being cleared as well as a descriptive string. The numFaults is also updated accordingly.

Read-Only Variable	numFaults
SNMP OID	1.3.6.1.4.1.50488.2.50.40.0
Syntax	Gauge32
Description	Total number of radio equipment faults currently active, across all RMU-managed nodes. Corresponds to the number of fault entries in the faultTable.

Table	faultTable
SNMP OID	1.3.6.1.4.1.50488.2.50.41
Description	List of fault entries. The number of faults is given by the value of numFaults.

12.7.4.1 Fault entry

Each Radio Equipment fault can be managed individually; they all have their own set of monitoring elements.

Table entry	faultEntry
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1
Indexes	faultBoardIndex, faultPortIndex, faultHopIndex, faultType
Description	Radio equipment fault entry containing management information applicable to a fault occurring on a radio equipment managed by the RMU. The values of faultBoardIndex, faultPortIndex and faultHopIndex uniquely defines the location of the faulty node in the fronthaul topology. The type of the fault is enumerated by the faultType value.

Read-Only Variable	faultBoardIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.1.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	Gauge32
Description	Device 'minor' identifier of the Swallow board to which the faulty node is connected.

Read-Only Variable	faultPortIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.2.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	Gauge32
Description	Index of the Swallow port to which the faulty node is connected (either directly or indirectly in the daisy-chaining case). This value also corresponds to the physically labeled index of the Swallow CPRI port (e.g. SFP_1 if faultPortIndex is 1).

Read-Only Variable	faultHopIndex
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.3.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	Gauge32
Description	Number of hops (networking radio equipments), in CPRI daisy-chaining topology between the Swallow CPRI master port and the faulty node's CPRI slave port. This index identifies the position of the faulty node in the daisy-chain for the Swallow port. If faultHopIndex is 0, then the faulty node is directly connected to the Swallow port.

Read-Only Variable	faultType
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.4.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	INTEGER (enumerated)
Description	Represents the type of the fault, among a few pre-determined values: ➢ extSupplyUndervolt(0) : Power supply input voltage is below acceptable range. ➢ overTemp(1) : Product temperate reaches pre-defined upper limit. ➢ digInOverdrive(2) : Digital signal level in transmit direction is above maximum acceptable value for the product. ➢ rfOutOverdrive(3) : RF output signal level is above maximum acceptable value for the product. ➢ txGainFail(4) : Tx gain cannot be adjusted to match required output power. ➢ rxGainFail(5) : Rx gain cannot be adjusted to match required signal level. ➢ vswrOutOfRange(6) : The voltage standing wave ratio at the antenna port has left acceptable range, return loss is outside acceptable value for the product.
CLI equiv.	swa:X/port:Y/node:Z/fault:U info

Read-Only Variable	faultProductName
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.5.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	DisplayString
Description	Product name of the faulty node. The string is a concatenation of the product part number, revision and serial number.
CLI equiv.	swa:X/port:Y/node:Z/fault:U info

Read-Only Variable	faultSeverity
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.6.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	INTEGER (enumerated)
Description	Represents the severity of the fault: ➢ warning(0) : A condition which may potentially lead to a fault situation, but does not cause any degradation at current time. The intention of this severity is to allow preventive action. ➢ degraded(1) : A degradation of performance, or a potential danger for the overall radio equipment health, without immediate failure. ➢ failed(2) : A definite failure to provide service, e.g. loss of a signal path or significant loss of output power.
CLI equiv.	swa:X/port:Y/node:Z/fault:U info

Read-Only Variable	faultDescr
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.7.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	DisplayString
Description	Node-generated string describing the fault with more details.
CLI equiv.	swa:X/port:Y/node:Z/fault:U info

Read-Only Variable	faultDuration
SNMP OID	1.3.6.1.4.1.50488.2.50.41.1.8.faultBoardIndex(X).faultPortIndex(Y).faultHopIndex(Z).faultType(U)
Syntax	TimeTicks
Description	The time (in hundredths of a second) since the fault became active.
CLI equiv.	swa:X/port:Y/node:Z/fault:U info

12.8 File Transfer Protocol (FTP)

The RMU offers a way to transfer files more easily to Radio Equipments, using FTP.
However, the FTP server does not reside inside the RMU. The actual FTP server can be installed separately on the host computer or any other computer available on the network that can be reached by Radio Equipments. The user can then define FTP "links" which are merely references to the FTP server location, credential and files location in the FTP server. Those links are then used by other commands such as `nodeSwUpgrade`.

Typically, the FTP links configuration is provided by the RMU XML configuration file. However, it is possible to dynamically change the FTP links definitions (such as the login credentials) using the SNMP/CLI user interface.

For instance, below Figure 61 shows how to add a new FTP link to the RMU.

```
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost ftpAddLink.0 s "myFTP"
AW2S-RMUv6-MIB::ftpAddLink.0 = STRING: "myFTP"
user@shuttle:~$ snmpget -v 2c -c public -m ALL localhost ftpAddLink.0
AW2S-RMUv6-MIB::ftpAddLink.0 = STRING: "ftp/link:1"
user@shuttle:~$
user@shuttle:~$
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost ftpLinkHost.1 a 192.168.0.100
Unexpected index type: 7 ftpLinkIndex 1
AW2S-RMUv6-MIB::ftpLinkHost.1 = IpAddress: 192.168.0.100
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost ftpLinkUserName.1 s username
Unexpected index type: 7 ftpLinkIndex 1
AW2S-RMUv6-MIB::ftpLinkUserName.1 = STRING: "username"
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost ftpLinkPassword.1 s password
Unexpected index type: 7 ftpLinkIndex 1
AW2S-RMUv6-MIB::ftpLinkPassword.1 = STRING: "password"
user@shuttle:~$ snmpset -v 2c -c private -m ALL localhost ftpLinkDirectory.1 s .
Unexpected index type: 7 ftpLinkIndex 1
AW2S-RMUv6-MIB::ftpLinkDirectory.1 = STRING: "."
user@shuttle:~$
```

Figure 61: Adding a FTP link

Giving the following `ftpLinkTable` content:

```
user@shuttle:~$ snmptable -v 2c -c public -m ALL -Cb localhost ftpLinkTable
SNMP table: AW2S-RMUv6-MIB::ftpLinkTable

 Index   ObjectName      Alias          Host   UserName   Password  Directory
     0 "ftp/link:0" "Releases"     127.0.0.1     "user" "default0" "releases"
     1 "ftp/link:1"    "myFTP" 192.168.0.100 "username" "password"         "."
user@shuttle:~$
```

Figure 62: FTP link table display

Read-Write Command	ftpAddLink
SNMP OID	1.3.6.1.4.1.50488.2.60.1.0
Syntax	AliasString
Description	Issuing a SET command on this field will add a new FTP link instance in the RMU. The new FTP link will be given the value of the SET command as alias. The alias will be usable posteriorly to reference to the FTP link. The FTP link entry is added to the `ftpLinkTable` with some default parameters.

	After successful creation, a GET command on this field will retrieve the object name of the new FTP link instance in the form `ftp/link:X` where `X` is the `ftpLinkIndex`. If the creation failed, a GET command on this field returns an empty string.
CLI equiv.	`ftp create-link <alias>`

Read-Write Command	**ftpRemoveLink**
SNMP OID	1.3.6.1.4.1.50488.2.60.2.0
Syntax	AliasString
Description	Issuing a SET command on this field will remove the FTP link instances for which the object name or alias correspond to the value of the SET command. The object name value is in the form `ftp/link:X` where `X` is the `ftpLinkIndex`. Relevant FTP links are removed from the `ftpLinkTable`. A 'Bad Value' failure of the SET command indicates that no FTP link instance matched the object name or alias. A GET command on this field always returns an empty string.
CLI equiv.	`ftp delete-link <name|alias>` `ftp delete-all-links`

Notification	**ftpLinkAdded**
SNMP OID	1.3.6.1.4.1.50488.2.60.12
Variables	ftpLinkObjectName, ftpLinkAlias, ftpNumLinks
Description	The `ftpLinkAdded` notification is emitted when a new FTP link entry has been added to the `ftpLinkTable` following a `ftpAddLink` command. The variable bindings include the new FTP link's `ftpLinkObjectName` and user-specified `ftpLinkAlias`. The `ftpNumLinks` is also updated accordingly.

Notification	**ftpLinkRemoved**
SNMP OID	1.3.6.1.4.1.50488.2.60.13
Variables	ftpLinkObjectName, ftpLinkAlias, ftpNumLinks
Description	The `ftpLinkRemoved` notification is emitted when a FTP link entry has been removed from the `ftpLinkTable` following a `ftpRemoveLink` command. The variable bindings include the removed FTP link's `ftpLinkObjectName` and user-specified `ftpLinkAlias`. The `ftpNumLinks` is also updated accordingly.

Read-Only Variable	**ftpNumLinks**
SNMP OID	1.3.6.1.4.1.50488.2.60.10.0
Syntax	Gauge32
Description	Number of FTP link entries declared in the `ftpLinkTable`.

Table	**ftpLinkTable**
SNMP OID	1.3.6.1.4.1.50488.2.60.11
Description	List of FTP link entries. The number of entries is given by the value of `ftpNumLinks`.

12.8.1 FTP link entry

Each declared FTP link can be managed individually; each one can refer to a specific FTP server with given login credentials. A FTP link entry will provide FTP server information to the Radio Equipment for a file transfer procedure.

Table entry	ftpLinkEntry
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1
Indexes	ftpLinkIndex
Description	FTP link entry containing management information applicable to an RMU FTP link.

Read-Only Variable	ftpLinkIndex
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.1.ftpLinkIndex(X)
Syntax	Gauge32
Description	Unique value for each FTP link used to index the list of FTP link entries.

Read-Only Variable	ftpLinkObjectName
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.2.ftpLinkIndex(X)
Syntax	AliasString
Description	The FTP link object name in the form ftp/link:X where X is the ftpLinkIndex.

Read-Write Variable	ftpLinkAlias
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.3.ftpLinkIndex(X)
Syntax	AliasString
Description	Alias for the FTP link usable posteriorly to reference it. This alias does not affect functionality in any way and can be chosen arbitrarily.
CLI equiv.	ftp/link:X alias [<alias>]

Read-Write Variable	ftpLinkHost
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.4.ftpLinkIndex(X)
Syntax	IpAddress
Description	IPv4 address of the machine on which the FTP server is hosted. If the FTP server is hosted locally, then the loopback address 127.0.0.1 may be used.
CLI equiv.	ftp/link:X host [<ftpIpAddr>]

Read-Write Variable	ftpLinkUserName
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.5.ftpLinkIndex(X)
Syntax	CredentialString
Description	User identifier required for logging into the FTP server.
CLI equiv.	ftp/link:X user [<ftpUserName>]

Read-Write Variable	ftpLinkPassword
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.6.ftpLinkIndex(X)
Syntax	CredentialString
Description	Password required for logging into the FTP server.
CLI equiv.	ftp/link:X password [<ftpPassword>]

Read-Write Variable	ftpLinkDirectory
SNMP OID	1.3.6.1.4.1.50488.2.60.11.1.7.ftpLinkIndex(X)
Syntax	DirNameString
Description	FTP server working directory for this FTP link. Files transferred using this FTP link as reference will be located in this directory on the FTP server. The value . means that the working directory is at the root of the FTP server directory structure.
CLI equiv.	ftp/link:X directory [<ftpDirectory>]

13 Integrated GPS receiver raw NMEA data

The board's integrated GPS receiver raw NMEA data is output continuously to Linux's serial console `/dev/ttySwallowGps0`, regardless of GPS synchronization being achieved or not.
This UART (9600 baud, 8 data bits, no parity bits, and one stop bit) console can be read by the user to retrieve real-time NMEA data as shown in Figure 63.

```
user@shuttle:~$ sudo cat /dev/ttySwallowGps0
$GPRMC,134416.000,A,4452.4996,N,00035.7696,W,0.2,0.6,171018,,,A*7E

$GPGGA,134416.000,4452.4996,N,00035.7696,W,1,07,2.8,043.37,M,49.6,M,,*73

$GNGSA,A,3,22,06,23,03,,,,,,,,,4.2,2.8,3.2*25

$GNGSA,A,3,69,68,85,,,,,,,,,,4.2,2.8,3.2*2D

$GPGSV,3,1,12,01,20,140,,02,01,316,,03,48,073,41,06,47,308,29*72

$GPGSV,3,2,12,07,04,170,,09,66,207,,11,01,154,,17,33,232,*76

$GPGSV,3,3,12,19,39,267,,22,29,086,31,23,79,099,50,31,08,031,*7E

$GLGSV,3,1,09,84,64,232,,83,19,175,,68,35,047,31,78,19,073,*69

$GLGSV,3,2,09,70,24,259,,77,12,023,,79,10,111,,69,68,327,34*6E

$GLGSV,3,3,09,85,35,323,23,,,,,,,,,,,,*54

$GPRMC,134417.000,A,4452.4998,N,00035.7694,W,0.1,0.0,171018,,,A*76

$GPGGA,134417.000,4452.4998,N,00035.7694,W,1,07,2.8,043.62,M,49.6,M,,*7E
```

Figure 63: Example GPS raw NMEA data

The supported NMEA sentences are:

- GGA (Fix information)
- GSA (Overall satellite data)
- GSV (Detailed satellite data)
- RMC (Recommended minimum data for GPS)
- VTG (Vector track and speed over ground)

Details on how to parse these sentences can be found at http://www.gpsinformation.org/dale/nmea.htm.
This NMEA data console is compatible for usage by external GPS parsing tools such as `gpsd` and/or `xgps` client.

Note: When using an active GPS antenna, GPS synchronization must be enabled in software in order to obtain valid fix information. This is due to the fact that the SYNC_IN SMA RF connector is biased to 3.3V only under this condition.

14 Annex A: PCIe and CPRI bandwidth

PCI-Express and CPRI bandwidths are limited resources that can impact the deployment of a system, depending on the wanted configuration and available hardware. This informative section is provided as a quick reference to estimate the available bandwidths, and their expected usage based on radio configuration.

14.1 PCIe available bandwidth

Because all data channels transit through the PCI-Express interface, it is particularly important to be aware of its available bandwidth. PCI-Express bandwidth is dependent on link speed, lane width and duplex traffic, but also on the underlying host hardware (such as, but not limited to, CPU, chipset, memory, other PCI devices in the system, negotiated payload size). Table 18 provides estimations of the available full-duplex bandwidth over the PCIe link.

Board type	PCIe link type	Theoretical bandwidth (Mbps)	Estimated available data bandwidth (Mbps)
A and B	Gen2 x4	16000	~8000
C, D, E and F	Gen3 x8	63000	~35000

Table 18: PCIe available full-duplex bandwidth estimation

The available PCI-Express data bandwidth on the host system can be measured thanks to an application that is automatically installed on the computer via the Software Installer package. Execute the command `swallow_bwtest` to do a measurement. Note that it is preferable to concede a small margin (e.g. 5%) on these values.

When using multiple boards in a single host, you can select on which board to run the bandwidth test by adding the board identifier (board index number, range is `[0...num_boards[`), parameter at the end of the command line, e.g.:
`swallow_bwtest 0`, or
`swallow_bwtest 1`.

14.2 CPRI available bandwidth

Another limitation to consider is the CPRI bandwidth. In this case, the bandwidth is well known and is dependent on the CPRI line bit rate option in use. The actual bandwidth available for I/Q data can be derived from the line speed, by considering the encoding rate and the control words. Values can be retrieved from Table 19.

Line bit rate option	Line speed (Mbps)	Encoding	Available data bandwidth (Mbps)
1	614.4	8B10B	460.8
2	1228.8	8B10B	921.6
3	2457.6	8B10B	1843.2
4	3072	8B10B	2304
5	4915.2	8B10B	3686.4

6	6144	8B10B	4608
7	9830.4	8B10B	7372.8
8	10137.6	64B66B	9216

Table 19: CPRI available full-duplex bandwidth

14.3 Signal carrier used bandwidth

The used bandwidth for a signal carrier is known and is dependent on the carrier type (channel bandwidth) and I/Q compression parameter. Table 20 shows bandwidth usage for a single signal carrier, in any direction (downlink or uplink).

Signal carrier type	I/Q compression parameter	PCIe bandwidth usage (Mbps)	CPRI bandwidth usage (Mbps)
NB-IoT	none	122.88	115.2
LTE 1.4MHz	none	122.88	115.2
LTE 3MHz	none	122.88	115.2
LTE 5MHz	none	245.76	230.4
LTE 10MHz	none	491.52	460.8
	lossless	368.64	345.6
	full	368.64	230.4
LTE 15MHz	none	737.28	691.2
	lossless	737.28	691.2
	full	737.28	460.8
LTE 20MHz	none	983.04	921.6
	lossless	737.28	691.2
	full	737.28	460.8
NR 5MHz	none	245.76	230.4
NR 10MHz	none	491.52	460.8
	lossless	368.64	345.6
NR 15MHz	none	737.28	691.2
NR 20MHz	none	983.04	921.6
	lossless	737.28	691.2
NR 25MHz	none	983.04	921.6
NR 30MHz	none	1474.56	1382.4
NR 40MHz	none	1474.56	1382.4
NR 50MHz	none	1966.08	1843.2

NR 60MHz	none	2949.12	2764.8
NR 70MHz	none	2949.12	2764.8
NR 80MHz	none	2949.12	2764.8
NR 90MHz	none	3932.16	3686.4
NR 100MHz	none	3932.16	3686.4

Table 20: Signal carrier bandwidth usage

14.4 Example system bandwidth requirements

Considering above tables, it is possible to compute the required bandwidth over the PCIe link and the CPRI links, and conclude if the available bandwidth on these interfaces is sufficient.
For example, let's consider that the LTEENB is to be configured using three radio cells:

- First cell runs LTE 20MHz in MIMO 2x2:
  - No I/Q compression.
  - Routed through CPRI_0.
- Second cell runs LTE 20MHz in MIMO 4x4:
  - Lossless I/Q compression.
  - Routed through CPRI_0.
- Third cell runs LTE 10MHz in MIMO 8x2:
  - Full I/Q compression
  - Routed through CPRI_1.

To compute the bandwidth requirements, retrieve the used bandwidth on each interface for each cell independently, then add them together to obtain the total used bandwidth, as described below.

First cell used bandwidth computation
This cell carrier type is LTE 20MHz, and I/Q compression parameter is set to `none`. From Table 20 we can deduce that, for each signal path, the PCIe bandwidth usage is 983.04 Mbps and CPRI bandwidth usage is 921.6 Mbps.
Running in MIMO 2x2 configuration, there is 2 Tx signal paths, and 2 Rx signal paths, therefore:

$$Cell_0 \begin{cases} PCIe(Tx) = 2 * 983.04 = 1966.08\ Mbps \\ PCIe(Rx) = 2 * 983.04 = 1966.08\ Mbps \\ CPRI_0(Tx) = 2 * 921.6 = 1843.2\ Mbps \\ CPRI_0(Rx) = 2 * 921.6 = 1843.2\ Mbps \end{cases}$$

Second cell used bandwidth computation
This cell carrier type is LTE 20MHz, and I/Q compression parameter is set to `lossless`. From Table 20 we can deduce that, for each signal path, the PCIe bandwidth usage is 737.28 Mbps and CPRI bandwidth usage is 691.2 Mbps.
Running in MIMO 4x4 configuration, there is 4 Tx signal paths, and 4 Rx signal paths, therefore:

$$Cell_1 \begin{cases} PCIe(Tx) = 4 * 737.28 = 2949.12\ Mbps \\ PCIe(Rx) = 4 * 737.28 = 2949.12\ Mbps \\ CPRI_0(Tx) = 4 * 691.2 = 2764.8\ Mbps \\ CPRI_0(Rx) = 4 * 691.2 = 2764.8\ Mbps \end{cases}$$

Third cell used bandwidth computation

This cell carrier type is LTE 10MHz, and I/Q compression parameter is set to `full`. From Table 20 we can deduce that, for each signal path, the PCIe bandwidth usage is 368.64 Mbps and CPRI bandwidth usage is 230.4 Mbps.
Running in MIMO 8x2 configuration, there is 8 Tx signal paths, and 2 Rx signal paths, therefore:

$$Cell_2 \begin{cases} PCIe(Tx) = 8 * 368.64 = 2949.12\ Mbps \\ PCIe(Rx) = 2 * 368.64 = 737.28\ Mbps \\ CPRI_1(Tx) = 8 * 230.4 = 1843.2\ Mbps \\ CPRI_1(Rx) = 2 * 230.4 = 460.8\ Mbps \end{cases}$$

Total

$$Total \begin{cases} PCIe(Tx) = 1966.08 + 2949.12 + 2949.12 = \boxed{7864.32\ Mbps} \\ PCIe(Rx) = 1966.08 + 2949.12 + 737.28 = \boxed{5652.48\ Mbps} \\ CPRI_0(Tx) = 1843.2 + 2764.8 = \boxed{4608\ Mbps} \\ CPRI_0(Rx) = 1843.2 + 2764.8 = \boxed{4608\ Mbps} \\ CPRI_1(Tx) = \boxed{1843.2\ Mbps} \\ CPRI_1(Rx) = \boxed{460.8\ Mbps} \end{cases}$$

Conclusion

By cross-checking with Table 18, we can say that the Gen2 x4 PCIe link is likely to have sufficient bandwidth for this configuration, however due to 7864.32 Mbps required Tx bandwidth being close to the estimation, it should be checked against the measured available bandwidth values reported by the `swallow_bwtest` application.
Regarding CPRI bandwidths, we can determine the needed line bit rate options from Table 19. In our case we can infer that CPRI_0 must run at line bit rate option 6 or higher, and CPRI_1 must run at line bit rate option 3 or higher.

15 Annex B: Example RAN setup

This section provides an example Radio Access Network setup with the corresponding Swallow configuration file (`swallow.xml`). The intent is to help the user gain a better understanding of how the radio cells signal paths are mapped to/from the correct Radio Equipment, based on configuration files.

Suppose that we want, on a single eNodeB, the following radio configuration:

- First sector:
  - LTE 10MHz TDD in MIMO 2x2
  - In 3GPP band 41, DL center frequency 2593MHz
- Second sector:
  - Intra-band 2-CA LTE 20MHz FDD in MIMO 4x4
  - In 3GPP band 3, DL center frequencies 1840MHz and 1860 MHz (contiguous CA)

An example `enb.cfg` file for this configuration is shown in Figure 64.

enb.cfg

```
/* RF driver configuration */
include "swallow.cfg",

/* list of cells */
cell_list: [
  {
      rf_port: 0,
      cell_id: 0x01,
      tac: 0x0001,
      n_id_cell: 1,
      root_sequence_index: 204,
      dl_earfcn: 40620,                                   /* B41 @ 2593 MHz */
      n_antenna_dl: 2, n_antenna_ul: 2,                   /* MIMO 2x2 */
      n_rb_dl: 50, sib_sched_list: [ "sib23_rb50.asn" ],  /* LTE 10MHz */
      uldl_config: 1, sp_config: 6,
  },
  {
      rf_port: 1,
      cell_id: 0x02,
      tac: 0x0001,
      n_id_cell: 2,
      root_sequence_index: 28,
      dl_earfcn: 1550,                                    /* B3 @ 1840 MHz */
      n_antenna_dl: 4, n_antenna_ul: 4,                   /* MIMO 4x4 */
      n_rb_dl: 100, sib_sched_list: [ "sib23_rb100.asn" ], /* LTE 20MHz */
      scell_list: [ { cell_id: 0x03, cross_carrier_scheduling: false, }, ],
  },
  {
      rf_port: 2,
      cell_id: 0x03,
      tac: 0x0001,
      n_id_cell: 3,
      root_sequence_index: 82,
      dl_earfcn: 1750,                                    /* B3 @ 1860 MHz */
      n_antenna_dl: 4, n_antenna_ul: 4,                   /* MIMO 4x4 */
      n_rb_dl: 100, sib_sched_list: [ "sib23_rb100.asn" ], /* LTE 20MHz */
      scell_list: [ { cell_id: 0x02, cross_carrier_scheduling: false, }, ],
  },
],
```

Figure 64: Example RAN setup LTEENB configuration

Let's consider for this example that the Radio Equipment nodes are all AW2S Blackhawk Remote Radio Heads, they come with the following capabilities:

- Have one CPRI slave port and one CPRI master port.
  - Support line bit rate options 1 to 6.
  - Support routing of I/Q data between the slave and master ports.
- Have two physical antenna ports.
  - Support MIMO 2x2.
  - Output power 43dBm maximum per antenna port.
- Support all LTE channel bandwidths as well as I/Q compression.
- Support 2-CA.

We also consider for this example that the used Swallow board is of hardware type A, that is, it only supports up to CPRI line bit rate option 6. To reduce possible radio interferences with other eNodeB's, we will be using GPS synchronization (highly recommended when operating TDD transmission mode).

Among multiple possible fronthaul physical setups, an example is shown in Figure 65.

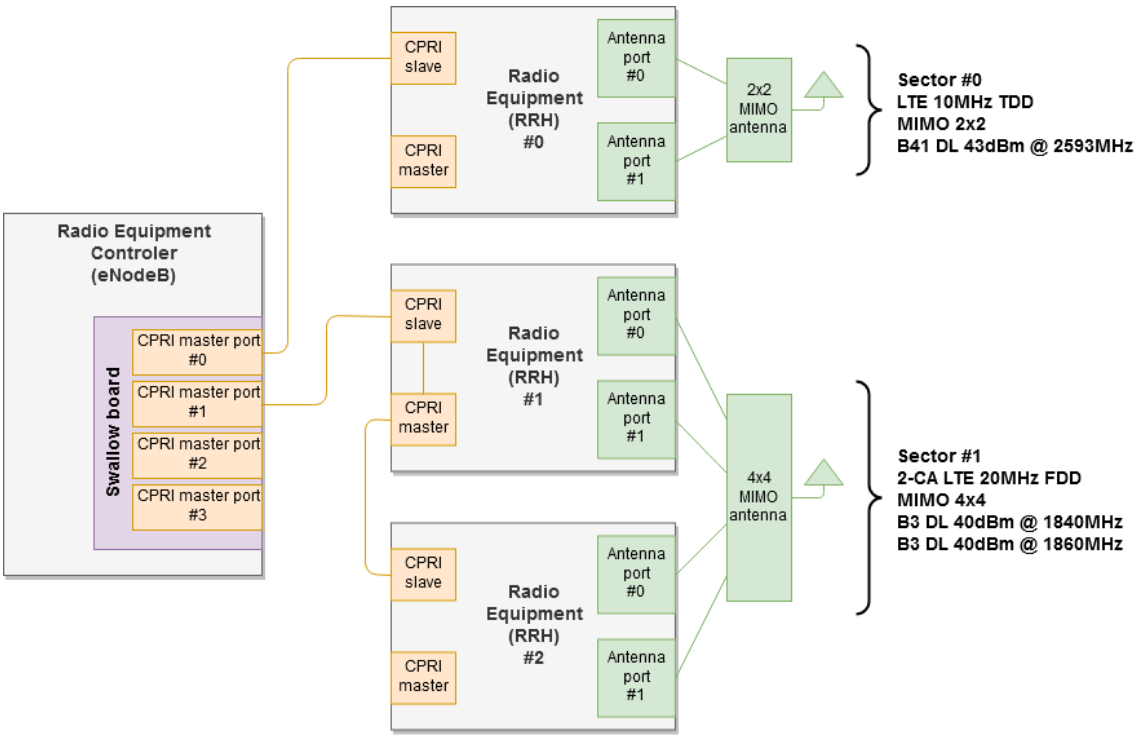


Figure 65: Example RAN setup physical configuration

While the RRHs only support up to MIMO 2x2, the Swallow board offers the possibility to combine two of these RRHs to operate in MIMO 4x4. Note that it would also be possible to chain the three RRHs and use only one Swallow CPRI master port, or have each RRH connected to its own Swallow master port.
With compatible User Equipment, we could expect 256-QAM downlink bitrates close to 800Mbps on sector #1.

Moving on to the Swallow configuration, it is usually done in three steps. First, define the required CPRI line speeds based on configured radio cells and physical setup. Next, configure the radio cells signal paths mapping. Finally, set the wanted maximum output power for the Tx signal paths.

The following part explains how the Swallow configuration file (`swallow.xml`) should be edited with regards to this example LTEENB setup and above physical configuration.

1) Define the required CPRI line speeds

For this step, we can refer to Annex A: PCIe and CPRI bandwidth as a quick reference.

Only the first cell (Band41 LTE 10MHz MIMO 2x2) is routed through Swallow CPRI master port #0. The required CPRI bandwidth is 921.6 Mbps, so any line bit rate option above or equal to 2 is sufficient.

The other two cells (2-CA Band3 LTE 20MHz MIMO 4x4) are both routed through Swallow CPRI master port #1. In this case the required CPRI bandwidth is 7372.8 Mbps, which is higher than the bandwidth available at the maximum CPRI line bit rate option supported by the board and RRHs (line bit rate option 6: 4608 Mbps). A solution is to use I/Q compression: with lossless compression the required bandwidth is reduced to 5529.6 Mbps – still too high – but with full compression the required bandwidth is further reduced to 3686.4 Mbps. Using full compression, any line bit rate option above or equal to 5 is sufficient.

2) Configure the radio cells signal paths mapping

The first cell runs in MIMO 2x2 so we will have to declare Tx and Rx channels with `id` ranging from 0 to 1. The other two cells run in MIMO 4x4 so we will have to declare Tx and Rx channels with `id` ranging from 0 to 3.

The first cell is straightforward, because the target RRH for its data channels is directly connected (not daisy-chained) to Swallow CPRI master port #0, we can define `master-port` as 0, and `hop-count` as 0 (no routing of data through a networking RE).
Then each signal path is mapped to its corresponding RRH physical antenna port (i.e. Tx0 go to RRH#0 physical antenna port #0, Tx1 go to RRH#0 physical antenna port #1).

For the other two cells, we have to take the daisy-chaining into consideration, as well as the mapping of MIMO 4x4 onto 2x2-capable RRHs.
Both RRH's being (directly or indirectly) connected to Swallow CPRI master port #1, we can define `master-port` as 1 for all the data channels. However, we will be routing signal paths with `id` ranging from 0 to 1 to the first RRH (`hop-count` is set to 0), and signal paths with `id` ranging from 2 to 3 to the second RRH (`hop-count` is set to 1 because data is going through one intermediate networking RE).
We can then map each signal path to its corresponding RRH physical antenna port (i.e. Tx0 go to RRH#1 physical antenna port #0, Tx1 go to RRH#1 physical antenna port #1, Tx2 go to RRH#2 physical antenna port #0, and Tx3 go to RRH#2 physical antenna port #1).

3) Set the wanted maximum output power for the Tx signal paths

Finally, we define the output powers for the Tx signal paths. For the first cell, we can use the RRH maximum output power (43 dBm) for both data channels. However, for the other two cells, due to the aggregation of two carriers on a single antenna port, we have to reduce the output power of all data channels by 3dB. In that case, the aggregation of two 40dBm carriers will net a total output power of 43dBm per antenna port.

The completed Swallow configuration file for this example RAN setup is provided in Figure 66.

swallow.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Swallow V6 LTEENB TRx PCIe configuration file -->

<swallow minor="0">

        <!-- CPU management -->
        <cpu wait-mode="poll" irq-interval-us="250"/>

        <!-- Synchronization scheme (ignored when using MRAT transceiver) -->
        <sync mode="gps" output="umts"/>

        <!-- CPRI master ports setup (ignored when using MRAT transceiver) -->
        <port id="0" cpri-line-speed="3"/>
        <port id="1" cpri-line-speed="5"/>
        <port id="2" cpri-line-speed="3"/>
        <port id="3" cpri-line-speed="3"/>

        <!-- Cells configuration -->
        <cell id="0">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="43.0"/>
                <rx id="0" master-port="0" hop-count="0" antport="0"/>
                <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="43.0"/>
                <rx id="1" master-port="0" hop-count="0" antport="1"/>
        </cell>
        <cell id="1">
                <iq-compression type="full" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="1" hop-count="0" antport="0" power-dBm="40.0"/>
                <rx id="0" master-port="1" hop-count="0" antport="0"/>
                <tx id="1" master-port="1" hop-count="0" antport="1" power-dBm="40.0"/>
                <rx id="1" master-port="1" hop-count="0" antport="1"/>
                <tx id="2" master-port="1" hop-count="1" antport="0" power-dBm="40.0"/>
                <rx id="2" master-port="1" hop-count="1" antport="0"/>
                <tx id="3" master-port="1" hop-count="1" antport="1" power-dBm="40.0"/>
                <rx id="3" master-port="1" hop-count="1" antport="1"/>
        </cell>
        <cell id="2">
                <iq-compression type="full" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="1" hop-count="0" antport="0" power-dBm="40.0"/>
                <rx id="0" master-port="1" hop-count="0" antport="0"/>
                <tx id="1" master-port="1" hop-count="0" antport="1" power-dBm="40.0"/>
                <rx id="1" master-port="1" hop-count="0" antport="1"/>
                <tx id="2" master-port="1" hop-count="1" antport="0" power-dBm="40.0"/>
                <rx id="2" master-port="1" hop-count="1" antport="0"/>
                <tx id="3" master-port="1" hop-count="1" antport="1" power-dBm="40.0"/>
                <rx id="3" master-port="1" hop-count="1" antport="1"/>
        </cell>
</swallow>
```

Figure 66: Example RAN setup Swallow configuration

16 Annex C: Example multi-board LTEENB configuration

Starting from Swallow software version 6.11, the Multi-RAT LTEENB transceiver (`swallow_mrat`) supports cell mapping across multiple Swallow boards within the same instance of LTEENB.

Configuring the Swallow M-RAT LTEENB transceiver for multi-board usage is done by declaring multiple Swallow boards in the Swallow configuration file (`swallow.xml`). For instance, let us consider that we have configured the Amarisoft LTEENB for two radio cells, and that we wish to map each cell to a different Swallow board. This can be done as seen in Figure 67.

swallow.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Swallow V6 LTEENB TRx PCIe configuration file -->

<!-- First Swallow board -->
<swallow minor="0">

        <!-- Cells configuration -->
        <cell id="0">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="30.0"/>
                <rx id="0" master-port="0" hop-count="0" antport="0"/>
                <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="30.0"/>
                <rx id="1" master-port="0" hop-count="0" antport="1"/>
        </cell>

</swallow>

<!-- Second Swallow board -->
<swallow minor="1">

        <!-- Cells configuration -->
        <cell id="1">
                <iq-compression type="none" tx-sigma="7000" rx-sigma="4000"/>
                <tx id="0" master-port="0" hop-count="0" antport="0" power-dBm="30.0"/>
                <rx id="0" master-port="0" hop-count="0" antport="0"/>
                <tx id="1" master-port="0" hop-count="0" antport="1" power-dBm="30.0"/>
                <rx id="1" master-port="0" hop-count="0" antport="1"/>
        </cell>

</swallow>
```

Figure 67: Multiple Swallow boards mapping configuration file

The first LTEENB cell (`cell id="0"`) is declared under the first Swallow board (`swallow minor="0"`) while the second LTEENB cell (`cell id="1"`) is declared under the second Swallow board (`swallow minor="1"`), achieving the desired result where each cell is mapped on its own board.
This configuration can be extended to more boards and cells if needed (i.e. map two cells under one Swallow board, and three cells under another Swallow board, for a total of five cells), or the order reversed (i.e. first cell on second board and second cell and first board) by adjusting the `minor` and/or `id` parameters.

However, for the multi-board transceiver to operate correctly, the boards need to be synchronized to each other so that they share the same clock and time-base.
In such a system, one board needs to be configured to act as the clock/sync master, while the other boards are configured as clock/sync slaves (they synchronize themselves to the clock master).
The synchronization methods used by the boards are setup in the RMU, by editing the RMU configuration file (`rmu.xml`). For instance, Figure 68 shows how to configure all boards to use the UMTS synchronization method (sync input and output and both set to UMTS), except the first board which will use GPS synchronization and act as

the clock master. This means that all boards in the system will be synchronized to the GPS (directly or indirectly), as long as physical connectors are indeed connected to the boards SYNC_IN and SYNC_OUT ports, in a daisy-chaining manner.

rmu.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Radio Management Unit configuration file -->

<rmu>

        <!-- Swallow configuration -->
        <swallow minor="*">
                <sync mode="umts" output="umts"/>
                <port id="*" manage="true" line-speed="3" defragment="true"/>
        </swallow>
        <swallow minor="0">
                <sync mode="gps"/>
        </swallow>
</rmu>
```

Figure 68: Inter-board synchronization RMU configuration file

In that setup, the physical connections are thus done as follow:
`Board #0: SYNC_IN <= GPS antenna`
And, for subsequent boards (`N>0`):
`Board #N: SYNC_IN <= SYNC_OUT(board#N-1)`

Note: When the LTEENB application is started using a multi-board configuration, the driver will check that all boards are indeed synchronized before data transmission can begin. If boards are not synchronized, then the application startup is aborted with a suitable error message.

Important: The inter-board synchronization cables may not be disconnected while the LTEENB is running in multi-board mode. Doing so will cause boards to lose synchronization to each other and the LTEENB will likely show Tx Underflow and/or Rx Overflow errors. In that case, the only way to recover from this state is to reconnect synchronization cables and restart the LTEENB application when synchronization has been reestablished.

17 Annex D: Troubleshooting

This section describes common issues that the user may encounter when using the Swallow V6 LTEENB transceiver, and attempts to give solutions to these problems. It is recommended that the user reads these troubleshooting guidelines before contacting AW2S support.

The PCIe board does not show in `lspci` output.
This means that the PCIe board was not detected by the BIOS/Linux at boot time.
Usually, this happens on cold starts, when the BIOS enumerates the PCIe buses before the board has finished powering-up.
A hot reboot of the computer is likely to solve the issue.
Note that there are sometimes options in the BIOS to increases the PCIe subsystem enumeration delay.
If a hot reboot does not solve the issue, then we can expect a more critical problem such as hardware defect.

- Check that the PCIe connector (both on motherboard and card) is not damaged due to many plug-out and plug-in cycles, moving the board to another PCIe slot can also be a good test.
- Check that the board is correctly powered up (green/red LEDs should be lit up on the board and front panel) and correctly plugged into the PCIe slot.

The `LnkSta` report from `lspci` indicates slower speed and/or smaller link width than expected.
The board can still function properly but with reduced PCIe bandwidth.

- Check that the host system and PCIe connector supports the expected speed and link width.
- Check that the PCIe connector (both on motherboard and card) is not damaged due to many plug-out and plug-in cycles, moving the board to another PCIe slot can also be a good test.

I can see from driver log that Swallow driver is loaded (registered), but no board is probed.

- Check that the board can be seen in `sudo lspci -vv` output.

I installed/upgraded the Swallow software and/or firmware and now Linux boot crashes.
This can happen when attempting to use a very old firmware (e.g. V4) with a more recent software, in this case there is a major incompatibility that cannot be handled gracefully by software, causing a kernel crash.
To have Linux boot again, unplug the PCIe board from the computer, and install the matching software version before plugging back the board, then proceed with a board firmware upgrade if needed.

I had installed the Swallow software, but after a reboot, the driver does not seem to load anymore; the output of `dmesg |grep swallow` is empty.
Usually, this happens when the Linux kernel version has changed.
When installing Swallow software, the driver is built against the currently running kernel, and is thus linked to that version only. If the kernel is changed later in time (either voluntarily or due to an automatic upgrade), the driver is not loaded anymore.
To fix this issue, re-install Swallow software with the new kernel version.
It is recommended to de-activate automatic kernel upgrades (which is enabled by default on some Linux distributions) to prevent this issue. Another way is to force GRUB to select a specific kernel at startup.

LTEENB startup fails with error `[RF] could not open swallow: Could not open device.`

- Check that a Swallow board has been probed with command `dmesg |grep swallow`.

LTEENB or RMU starts successfully, but software does not seem to connect to Radio Equipment for configuration.

- Check that the Radio Equipment is correctly powered-up and the optical links are cabled. The LEDs of the board's SFP ports should be green indicating CPRI link is up, if not, also make sure that the CPRI line bit rate option is supported by the used SFP transceiver, fiber cable, and Radio Equipment.
- Check the network interfaces of the Swallow board are up and running with a DHCP server.
  - Linux command `sudo service swallow status` should indicate that the service is running.
  - Linux command `ifconfig` should show interfaces `swaXpY` are up.
  - LTEENB command `rf_info` should **not** show `DHCPX: Unavailable` states.
  - RMU command `info -rv` should **not** show `Errored` statuses.
  - If any of the above is incorrect, try a `sudo service swallow restart` and/or `sudo service rmu restart`, or consider reinstalling Swallow software and checking the installation log.

LTEENB shows many Tx underflow and/or Rx overflow messages.

LTEENB is a real-time application that can require high CPU performance and low-latency kernels. CPU usage scales proportionally with channel bandwidth and the number of radio cells and antenna paths that were configured.
High CPU usage, latency and excessive PCIe bandwidth usage can cause overflows and underflows of the PCIe board's internal buffers, this causes discontinuities in the signal and may impede performance and stability. To help with this issue:

- Check that there is sufficient PCIe bandwidth for the wanted configuration. Refer to section Annex A: PCIe and CPRI bandwidth.
- Make sure no other unneeded application runs on the computer.
- Use the computer in command line only (no graphical interface).

More information on this issue can be found in Amarisoft's LTEENB documentation.
Also note that for Amarisoft release 2018-04-01 and newer, there is a known incompatibility with the 3.14.25-rt22 Linux kernel which may cause similar issues. In this case it is recommended to upgrade the kernel then re-install Swallow software.

My issue does not seem to be covered in this troubleshooting section, what can I do?

In this case, a support ticket should be opened at the following Redmine: https://supportaw2s.serma.com. For quicker and more efficient support, please provide the following in your ticket:

- A detailed explanation of your issue, what is (not) happening, and when did the issue arise.
- A short description of your host computer (CPU, Linux distribution and kernel version).
- The board's hardware type, firmware and software versions you are using.
- The output of `sudo lspci -vv` and `dmesg |grep swallow` and `sudo service swallow status` commands.
- If available and relevant, the LTEENB/LTEUE configuration files that you use (*`enb.cfg`* or *`ue.cfg`*, *`swallow.cfg`* and *`swallow.xml`*), as well as the LTEENB application console logs, the output of LTEENB command `rf_info`.
- If available and relevant, the RMU configuration file (`rmu.xml`), as well as the RMU application log files (located by default in `/var/log/rmu`), and the output of RMU command `info -rv`.
- The more logs, the better.

Please note that support may be limited by commercial agreement.

SENSECAP

LoRaWAN Gateway and Wireless Sensor User Guide

How to Work with 3rd-party Standard LoRaWAN Gateway or TTN Server

Version: V1.2

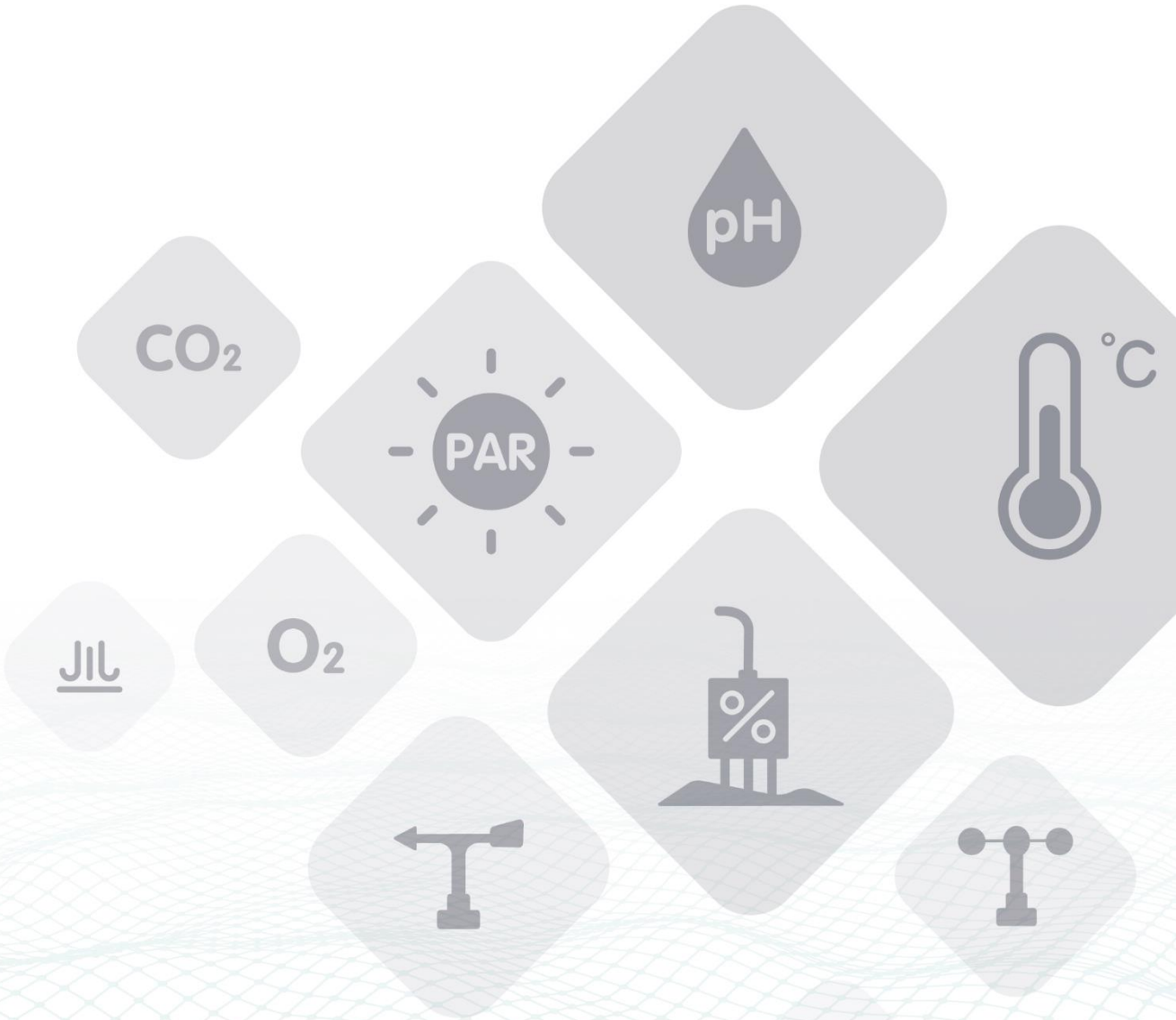


Table of Contents

1 Product Introduction

SenseCAP is an industrial wireless sensor network that integrates easy-to-deploy hardware and data API services, enabling low-power, long-distance environmental data collection. SenseCAP includes several versions, such as LoRaWAN, LoRaPP, etc.

SenseCAP LoRaWAN version products include LoRaWAN Gateways and Sensor Nodes. Based on the LoRaWAN protocol, it can realize one-to-many, long-distance networking and bilateral communication. The LoRaWAN Gateway supports Ethernet and 4G. The Sensor Node is powered by a high-capacity battery that lasts up to 3 years (if uploading data once every hour). It also supports hot-swap, making it easy for maintenance and upgrading.

Main Features:

- Gateway: High-performance Cortex A8 1GHz processor
- Gateway uses multiple methods to connect to the Internet: 4G and Ethernet
- Gateway supports third-party TTN account and server
- Sensors support LoRaWAN v1.0.2 protocol and are suitable for standard LoRaWAN Gateway
- Super long-distance communication: 10km in the line-of-sight scenario, 2km in the urban scenario
- Industrial protection rating IP66-rated enclosure, suitable for the outdoor environment at -40℃~70℃
- Easy-to-deploy, enabling people without engineering background to install the devices quickly

LoRaWAN Gateway:

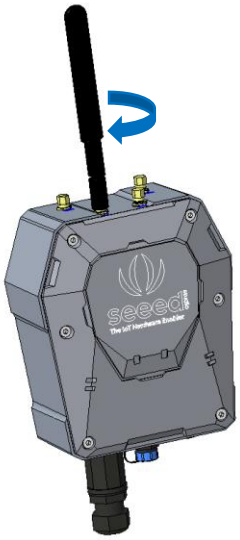
LoRaWAN Sensor Node:

2 Gateway Network Configuration

2.1 The gateway connects to the Internet

2.1.1 Installing Antenna

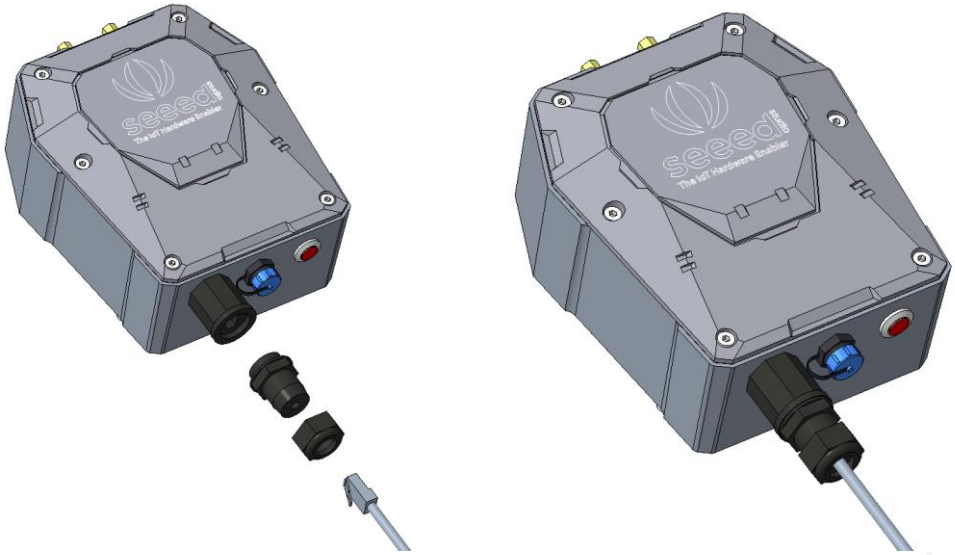
Screw clockwise to install the 4G and LoRa antennas onto the gateway.



2.1.2 Connecting to the Internet

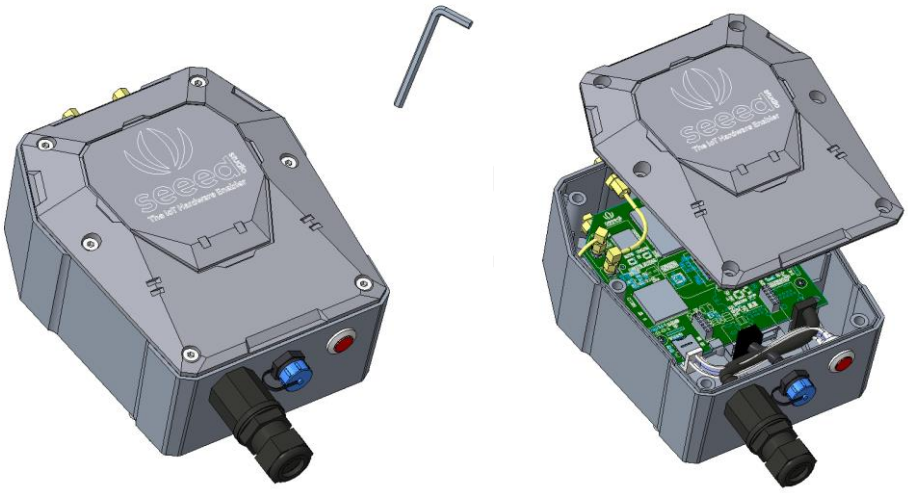
There are two ways to connect to the Internet. Choose the one that works for you.

(1) Connecting to Ethernet Cable
Unscrew to open the protection cap, plug the Ethernet cable through the cap and then into the Ethernet port. Screw to fasten this part.

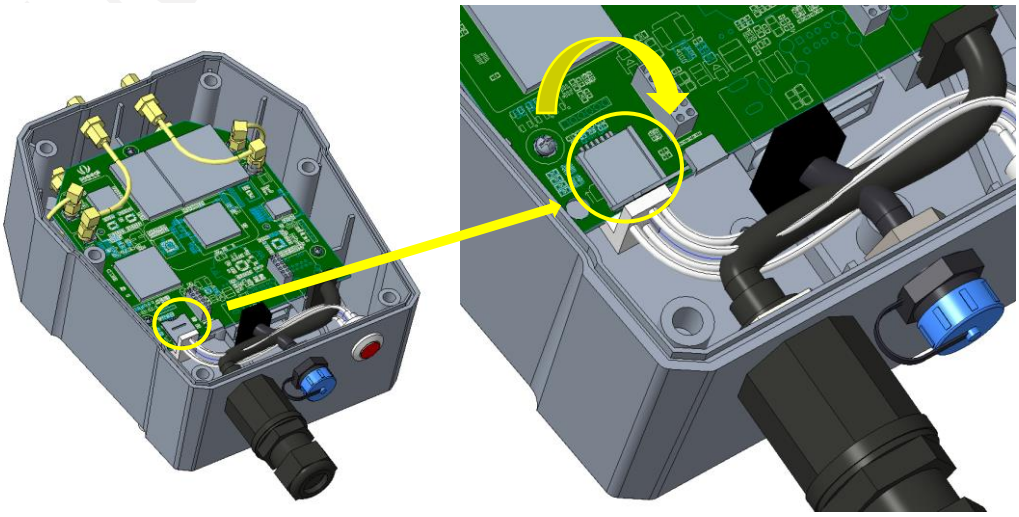


(2) Connecting to 4G

Use the hex key (included in the package) to unscrew the 6 screws and open the lid.

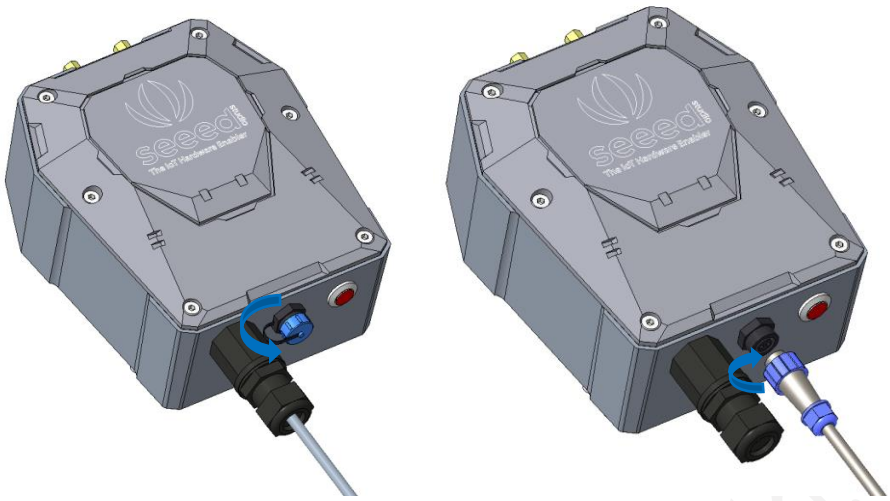


Swipe downward to open the SIM card socket, insert the Micro SIM card and swipe upward to lock the SIM card socket. Make sure it is installed correctly and close the lid with the screws.



2.1.3 Connecting to Power Cable

Unscrew to take off the power cap, plug in the extension cord and screw to fasten it onto the gateway. The other end of the extension cord is connected to the power adapter.

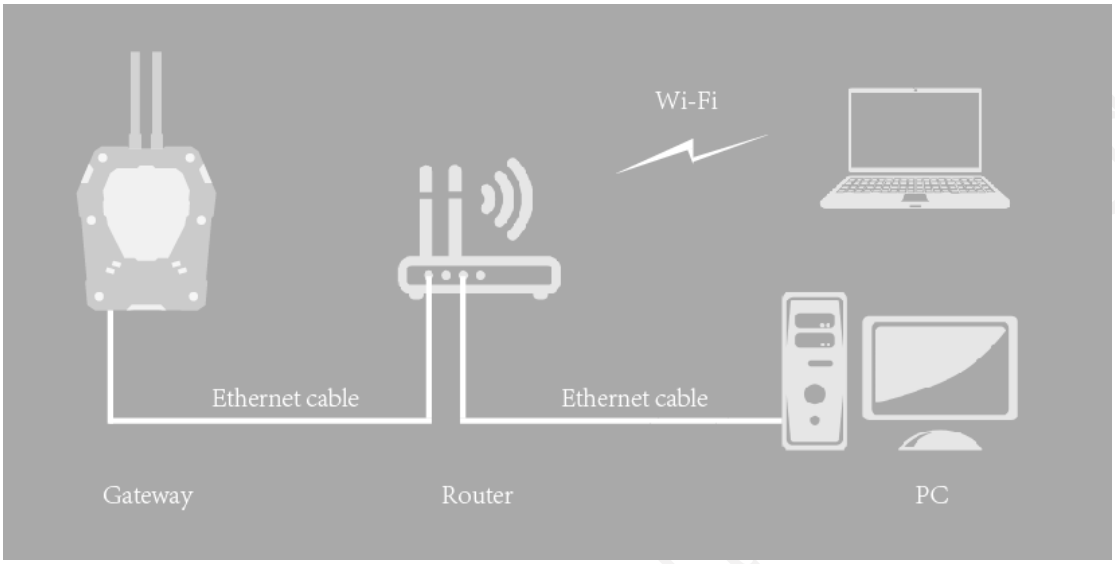


Notice: Make sure all antennas are correctly installed before powering on the gateway. Please note the device should be POWERED OFF when installing the antenna, or the antenna circuits might be damaged.

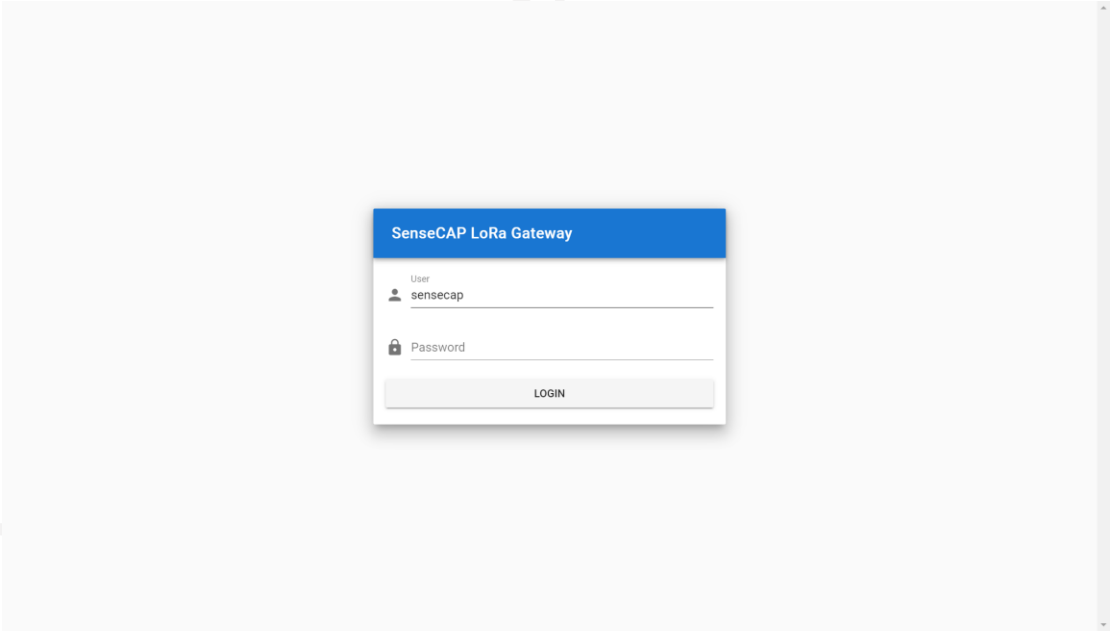
2.1.4 The Function of the Red LED

2.2 Setting the APN

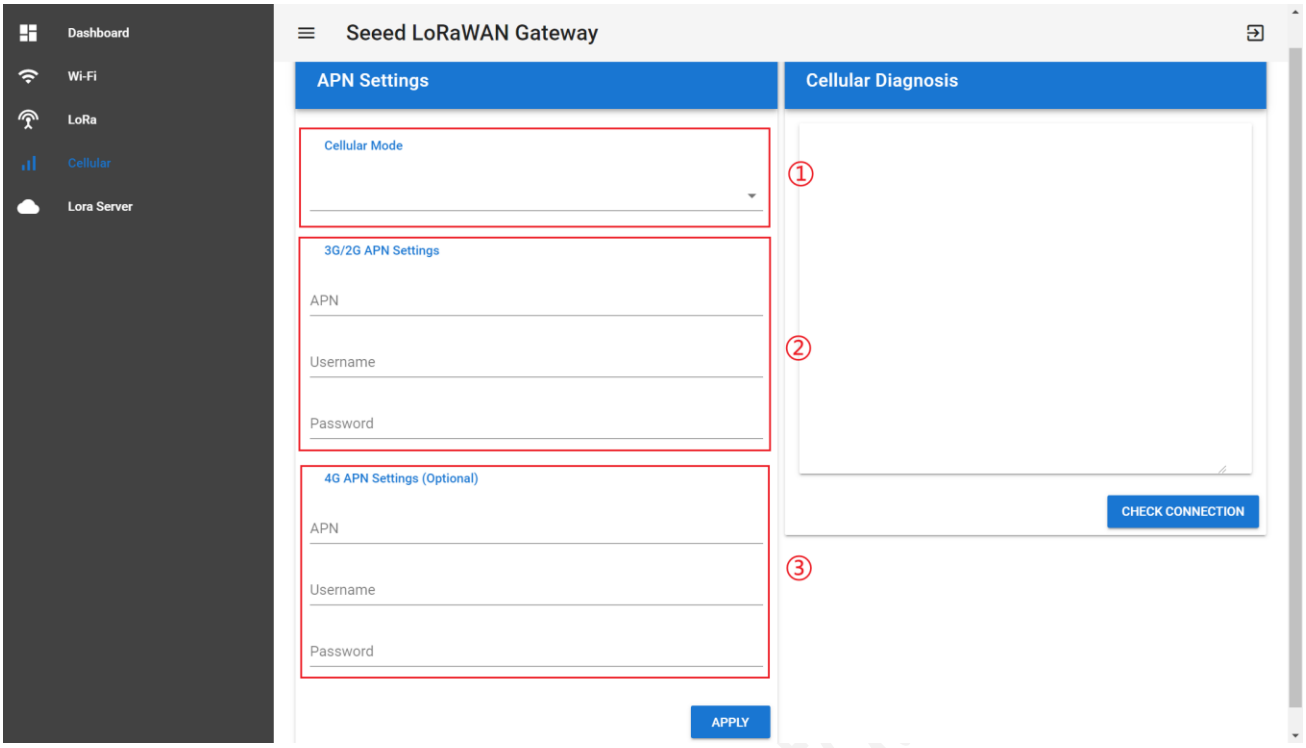
Prepare a router, and the network connection is shown in the figure:



(1) Check the IP of "sensecap" in the background of the router.
(2) Enter IP in the browser: IP:8000
If the IP is 192.168.1.1, enter 192.168.1.1:8000



(3) User: sensecap
Password: sensecap!!!
(4) Click the “Cellular” button.

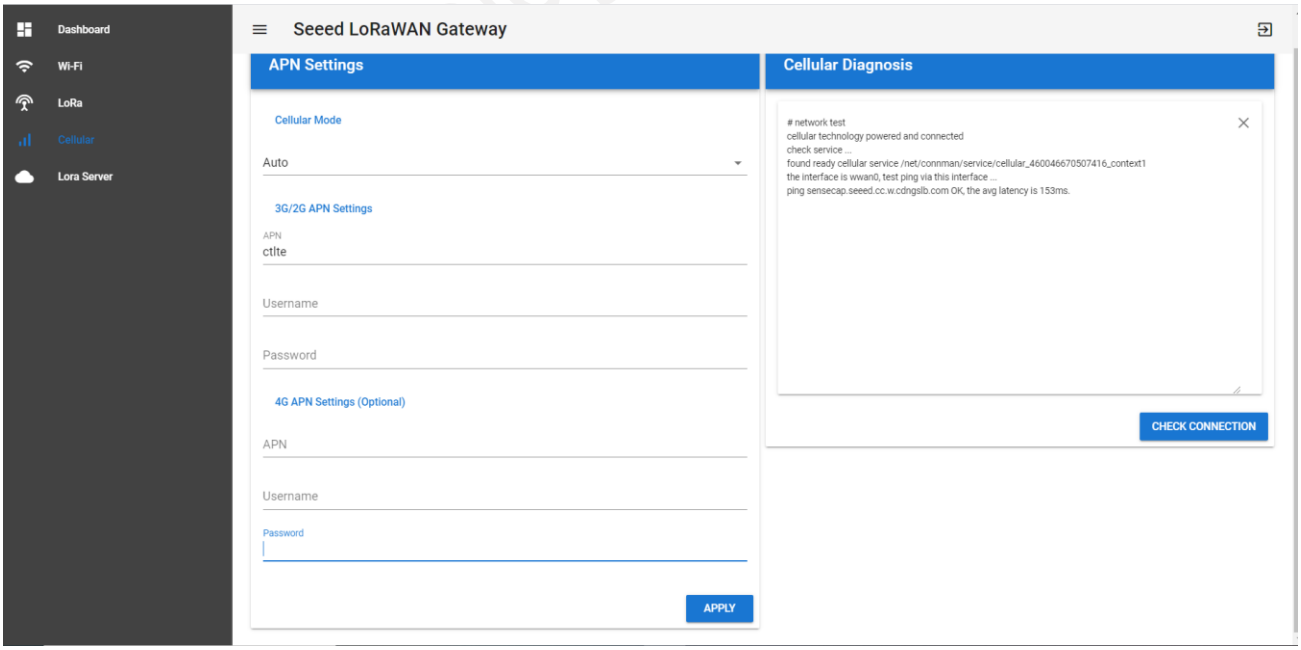


① Cellular Mode: AUTO(default), Gateway automatically selects mode.

② 3G/2G APN Settings: when the mode is 3G/2G, the APN information of SIM card operator needs to be filled in.

③ 4G APN Settings: optional.

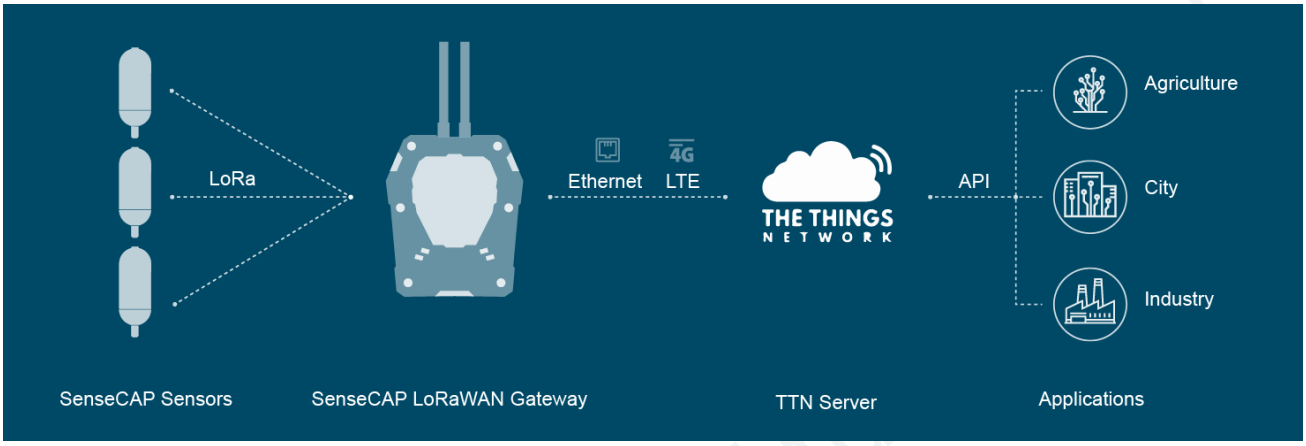
(5) Click “APPLY”. Then “CHECK CONNECTION”, if return “cellular technology powered and connected”, it means ok.



3 Add Gateway to User's TTN Server

The SenseCAP LoRaWAN Gateway supports connecting to the user's own The Things Network account and server.

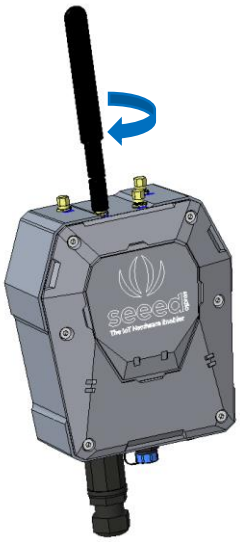
Learn more about TTN: https://www.thethingsnetwork.org/docs/



3.1 Gateway Network Configuration

3.1.1 Installing Antenna

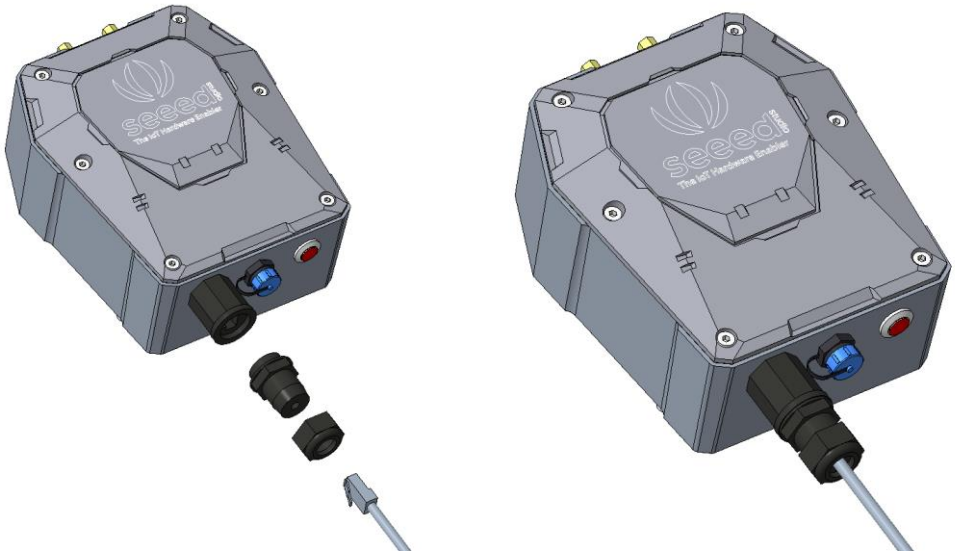
Screw clockwise to install the 4G and LoRa antennas onto the gateway.



3.1.2 Connecting to the Internet

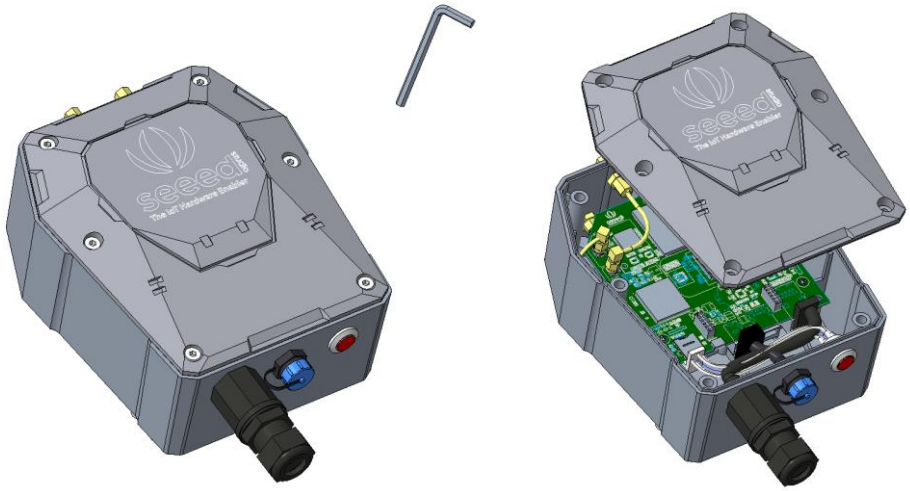
There are two ways to connect to the Internet. Choose the one that works for you.

(3) Connecting to Ethernet Cable
Unscrew to open the protection cap, plug the Ethernet cable through the cap and then into the Ethernet port. Screw to fasten this part.

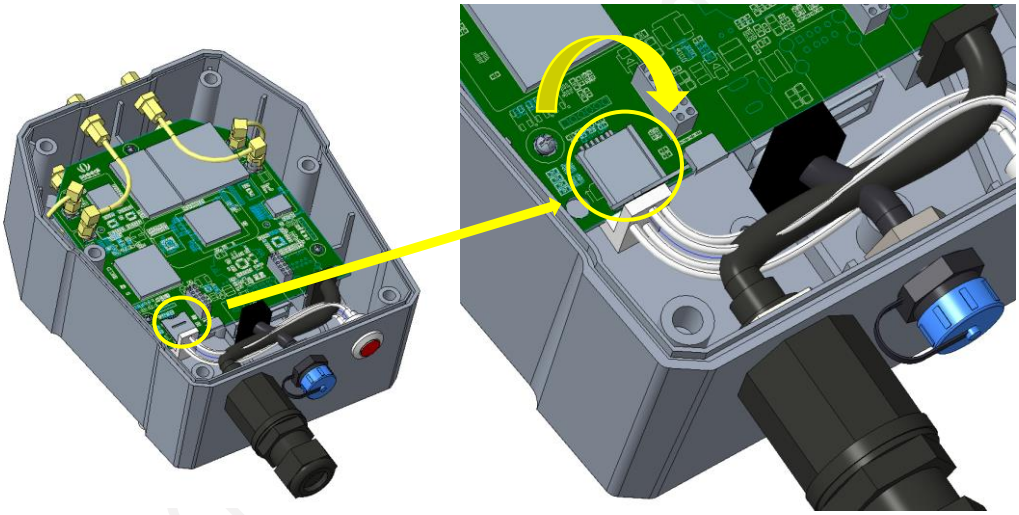


(4) Connecting to 4G

Use the hex key (included in the package) to unscrew the 6 screws and open the lid.

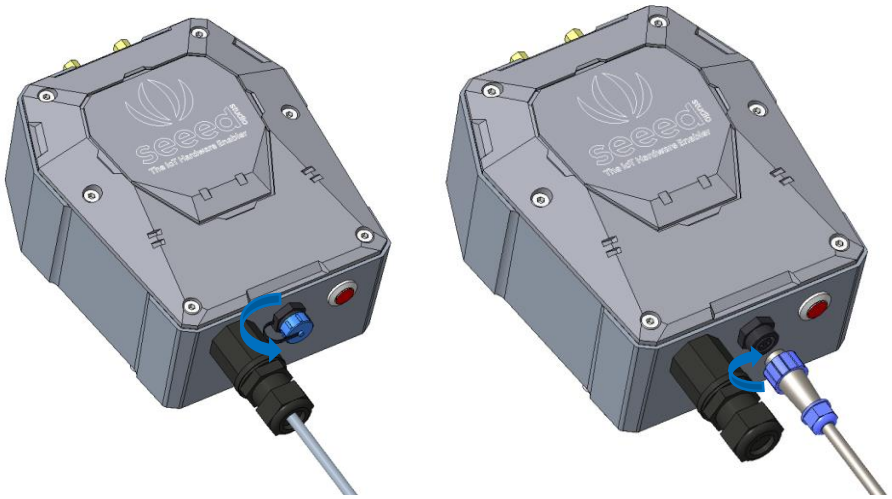


Swipe downward to open the SIM card socket, insert the Micro SIM card and swipe upward to lock the SIM card socket. Make sure it is installed correctly and close the lid with the screws.



3.1.3 Connecting to Power Cable

Unscrew to take off the power cap, plug in the extension cord and screw to fasten it onto the gateway. The other end of the extension cord is connected to the power adapter.

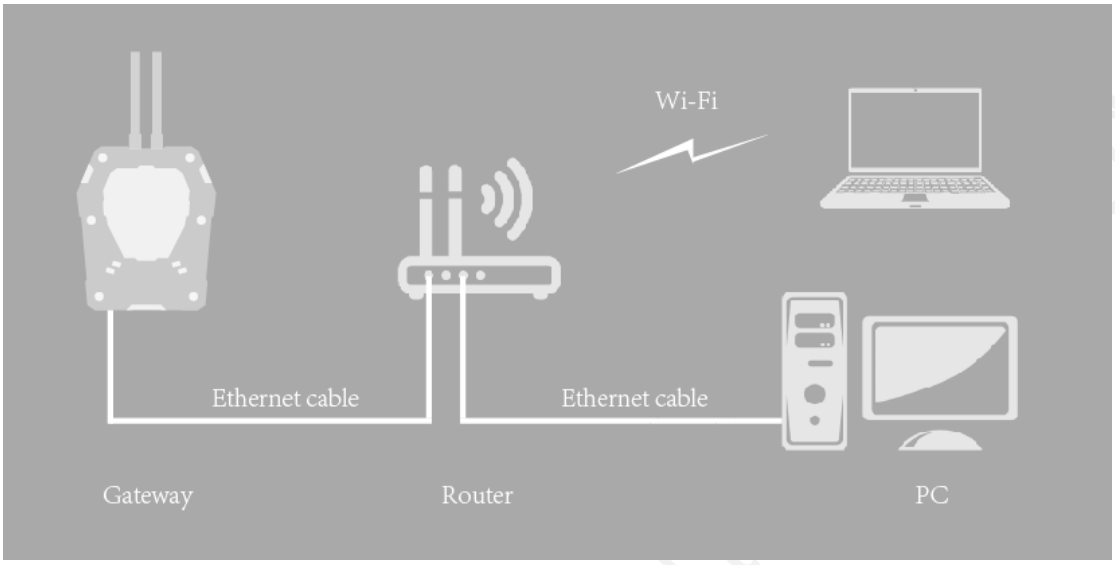


Notice: Make sure all antennas are correctly installed before powering on the gateway. Please note the device should be POWERED OFF when installing the antenna, or the antenna circuits might be damaged.

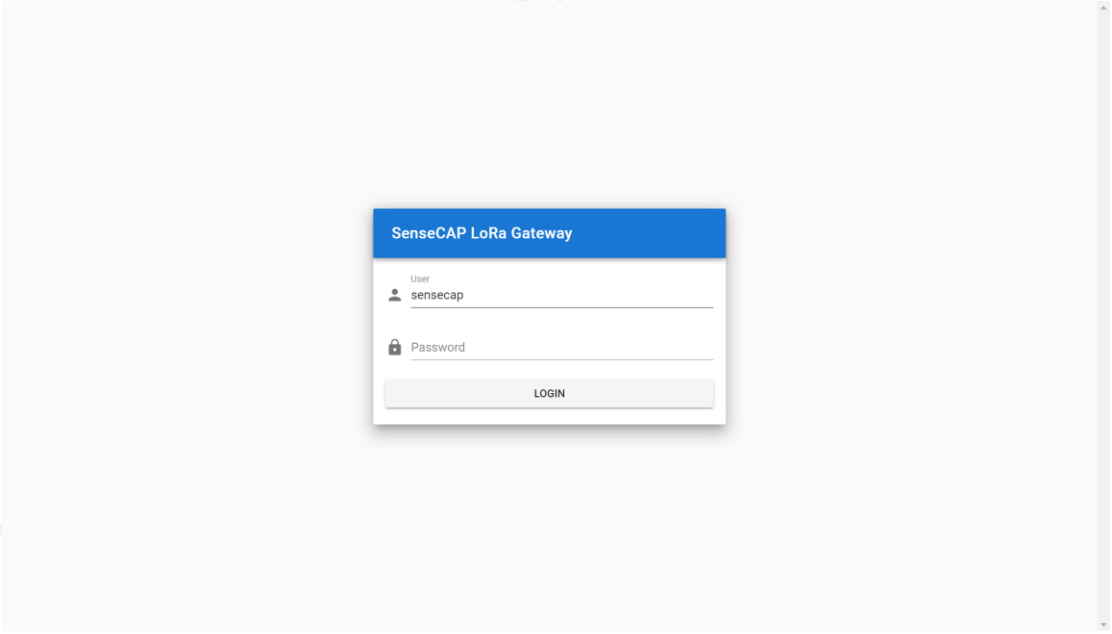
3.1.4 The Function of the Red LED

3.2 Setting the Gateway Service Address

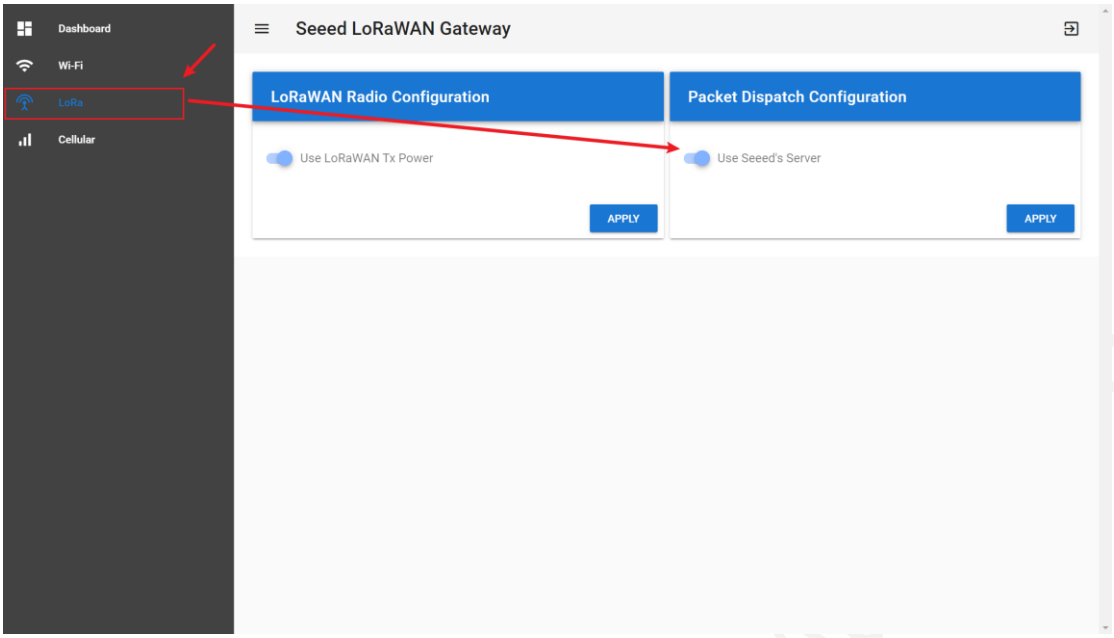
Prepare a router, and the network connection is shown in the figure:



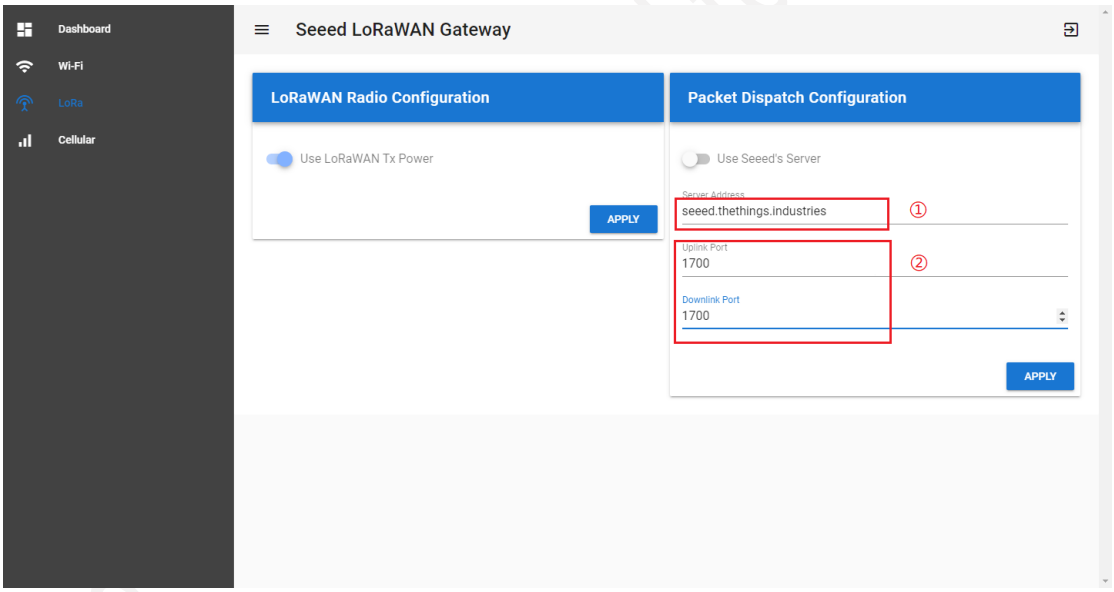
(6) Check the IP of "sensecap" in the background of the router.
(7) Enter IP in the browser: IP:8000
If the IP is 192.168.1.1, enter 192.168.1.1:8000



(8) User: sensecap
Password: sensecap!!!
(9) LoRa→Use Seeed's Server→Off Button



(10)



① Server Address: Please input your Server Address.

Refer to the table or website: https://www.thethingsnetwork.org/docs/gateways/packet-forwarder/semtech-udp.html#router-addresses

Router address	Region
router.eu.thethings.network	EU 433 and EU 863-870
router.us.thethings.network	US 902-928
router.cn.thethings.network	China 470-510 and 779-787
router.as.thethings.network	Southeast Asia 923 MHz

router.as1.thethings.network	Southeast Asia 920-923 MHz
router.as2.thethings.network	Southeast Asia 923-925 MHz
router.kr.thethings.network	Korea 920-923 MHz
router.jp.thethings.network	Japan 923-925 MHz (with EIRP cap according to Japanese regulations)
thethings.meshed.com.au	Australia 915-928 MHz
as923.thethings.meshed.com.au	Australia (Southeast Asia 923MHz frequency plan)
ttn.opennetworkinfrastructure.org	Switzerland (EU 433 and EU 863-870)

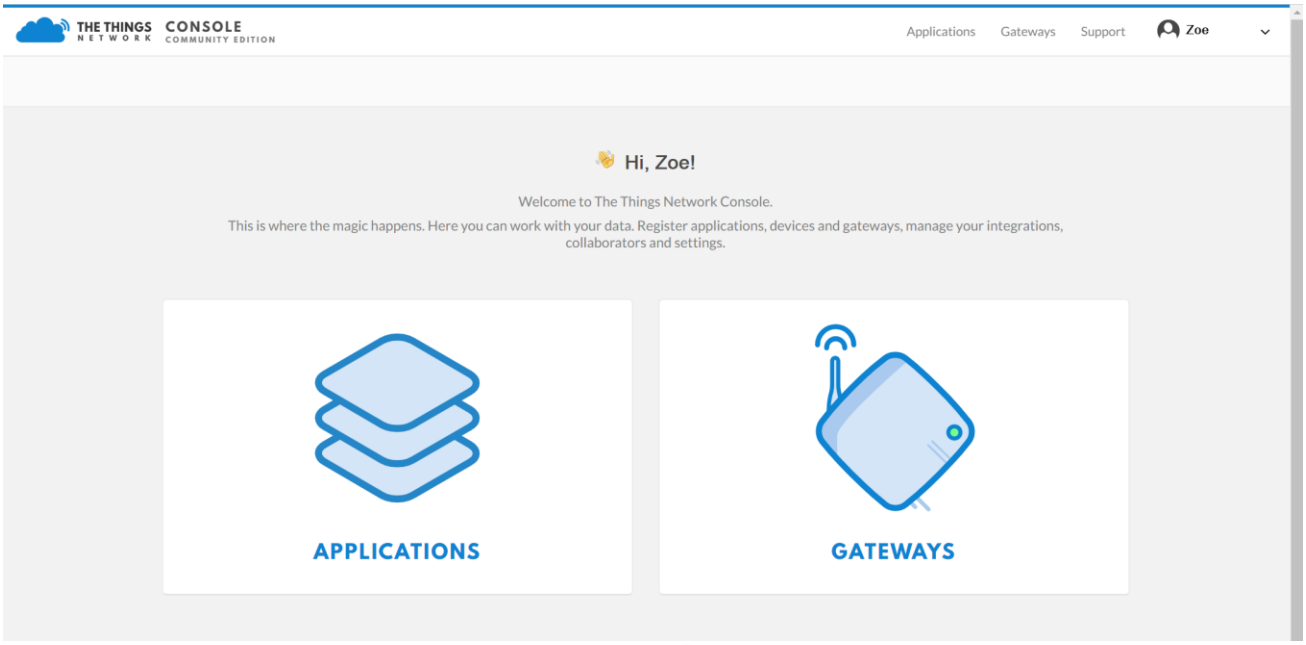
② Uplink / Downlink Port (default): 1700

(11) APPLY.

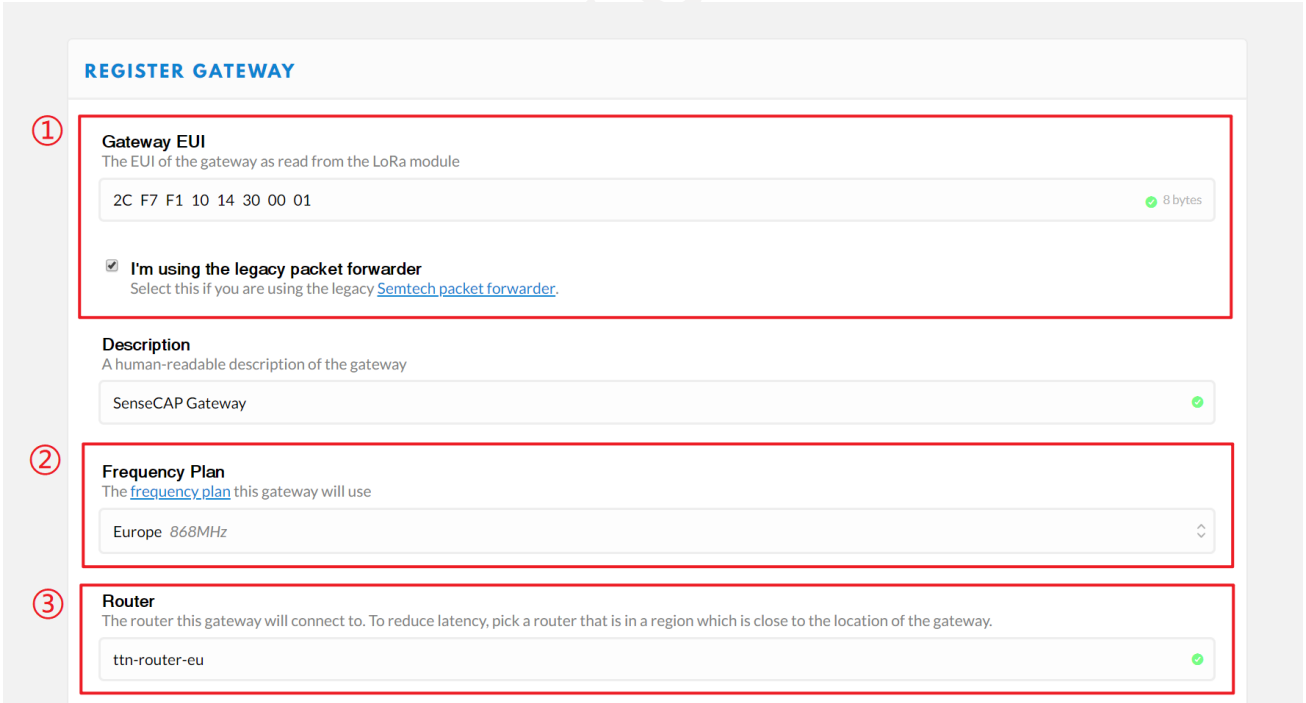
3.3 Gateway Registration on TTN

TTN website: https://www.thethingsnetwork.org

(1) Follow the instruction to create your account, and access “Console”.



(2) Register Gateway

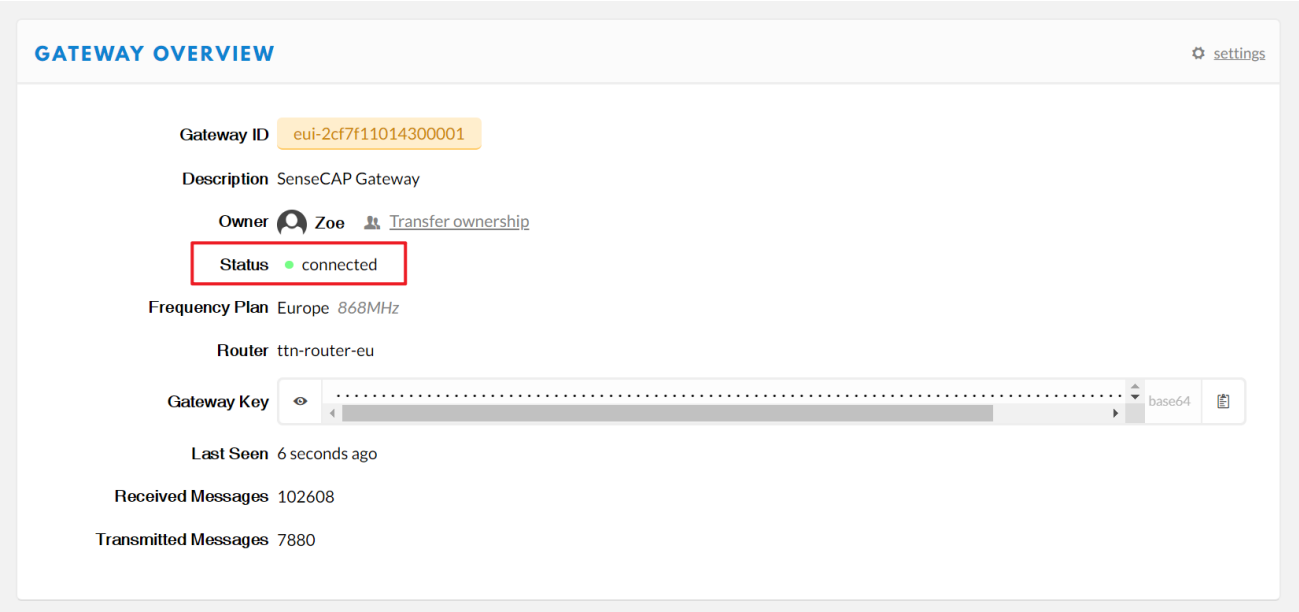


① Gateway EUI: View the labels on the gateway.
Select ‘I’m using the legacy packet forwarder’.

② Frequency Plan: View the labels on the gateway.

③ Router: Select the router that is right for you.

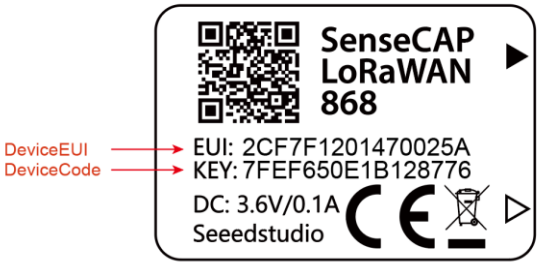
④ Register.
Gateway Status displays connected, indicating successful registration.



4 Add Sensor Node to User's TTN Server

4.1 Get Node's EUI and Key

(1) DeviceEUI and DeviceCode is on the SenseCAP product label.



(2) SenseCAP sensor device's AppEUI and AppKey have been flash into the device by Seeed. Use HTTP API to retrieve App EUI and App Key. You can use browser to issue an HTTP GET request.

Curl:

```
https://sensecap.seeed.cc/makerapi/device/view_device_info?nodeEui=2CF7F12014700297&deviceCode=34BF25920A4EFBF4
```

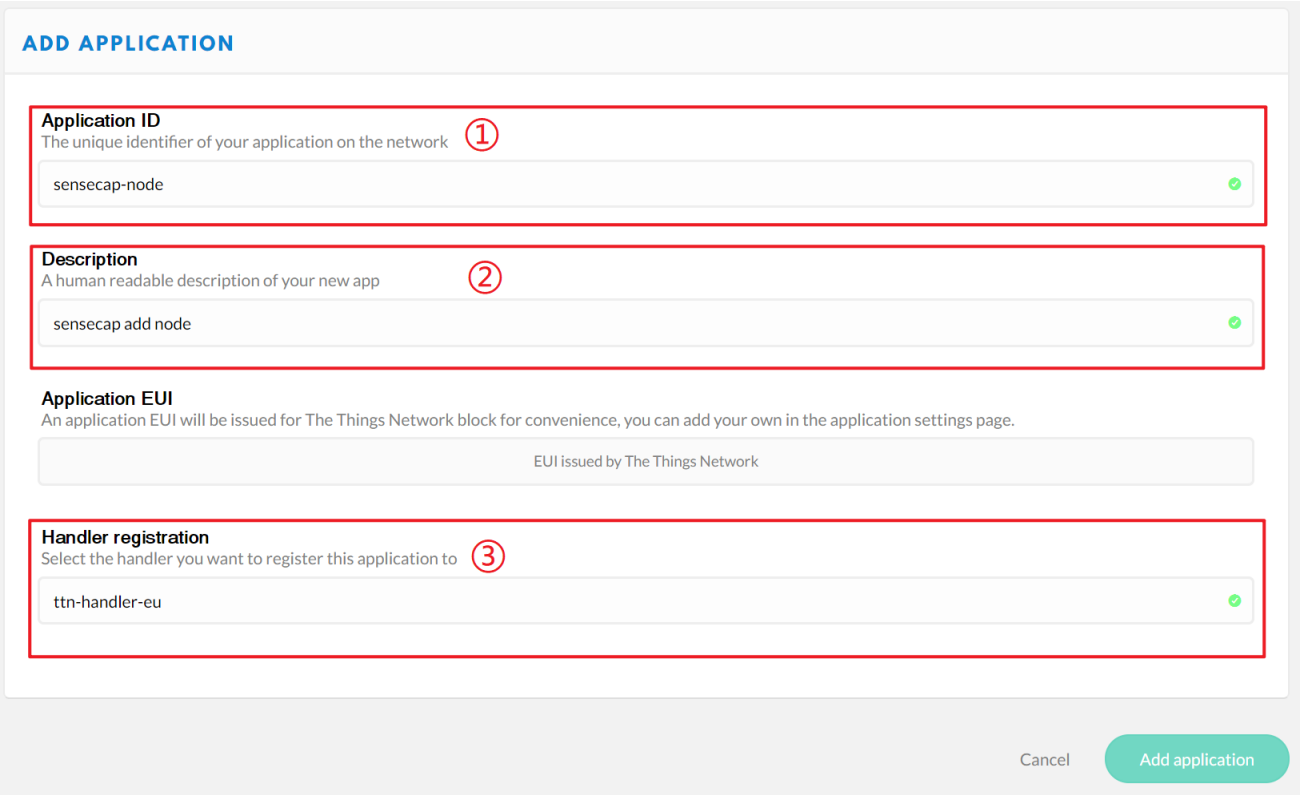
In the API, replace the DeviceEUI and deviceCode with your own DeviceEUI and DeviceCode respectively. And you will get the following response.

```
{
  "code": "0",
  "data": {
    "nodeEui": "2CF7F12014700297",
    "deviceCode": "34BF25920A4EFBF4",
    "lorawanInformation": {
      "dev_eui": "2CF7F12014700297",
      "app_eui": "8000000000000006",
      "app_key": "6FD0EF47CBC6E00F1921A08C2E94E8E5"
    }
  },
  "time": 0.019
}
```

4.2 Add Application and AppEUI

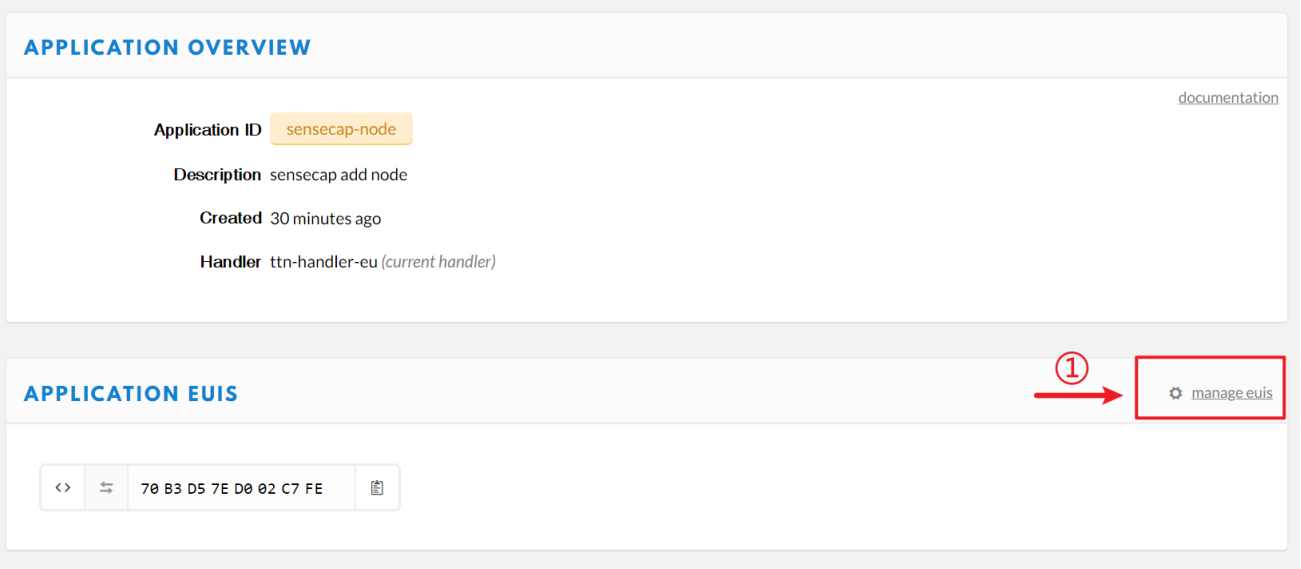
(1) TTN console → Application → Add application

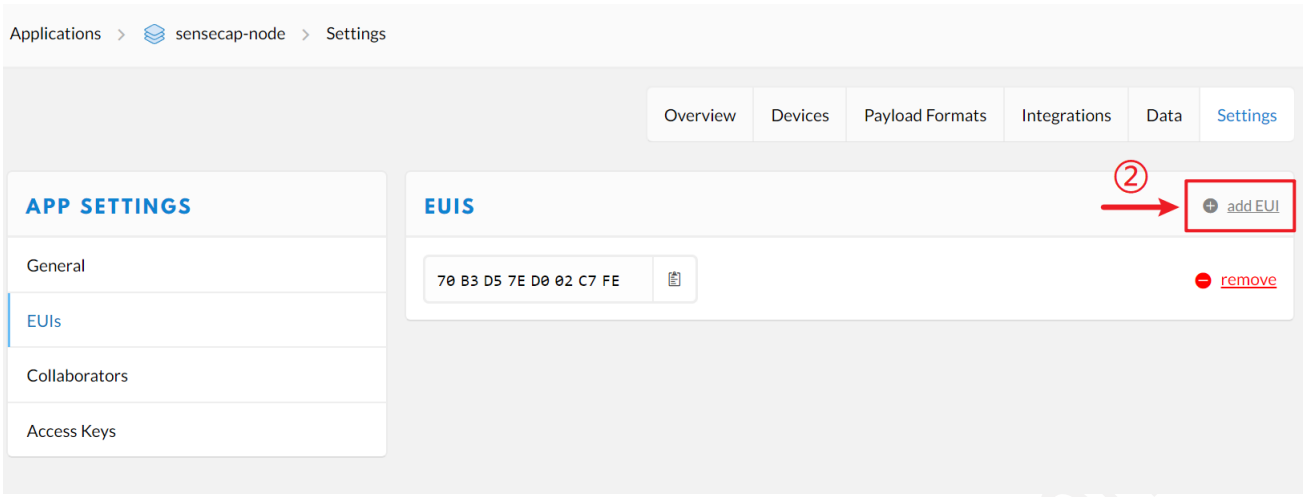
(2)

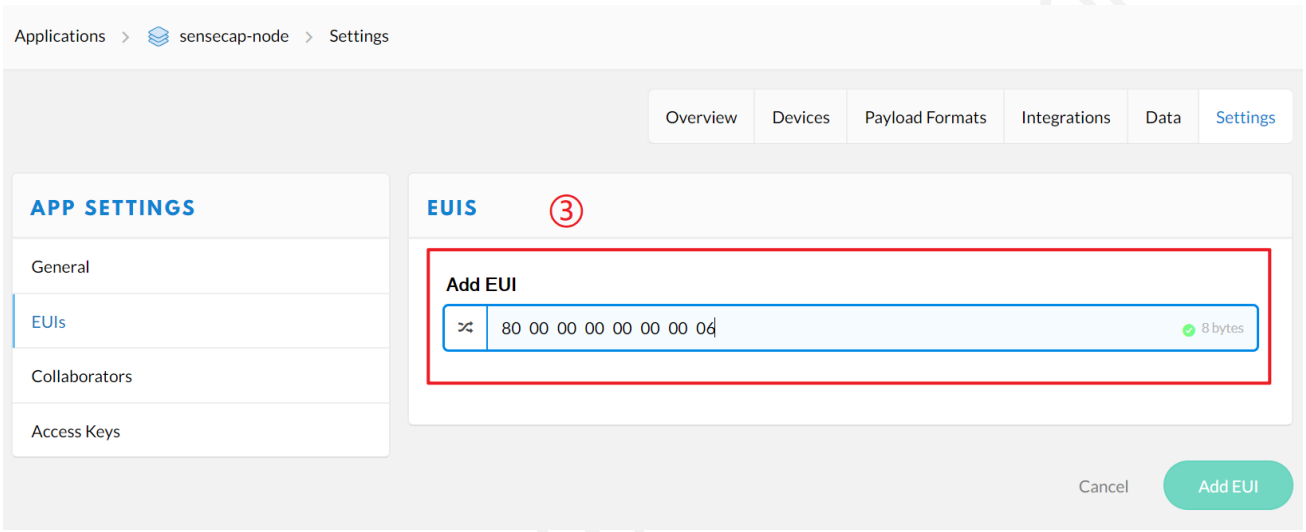


① Application ID: Enter a unique name.
② Description: Enter a description.
③ Handler registration: Select the same handler as the gateway router.
④ Add application.

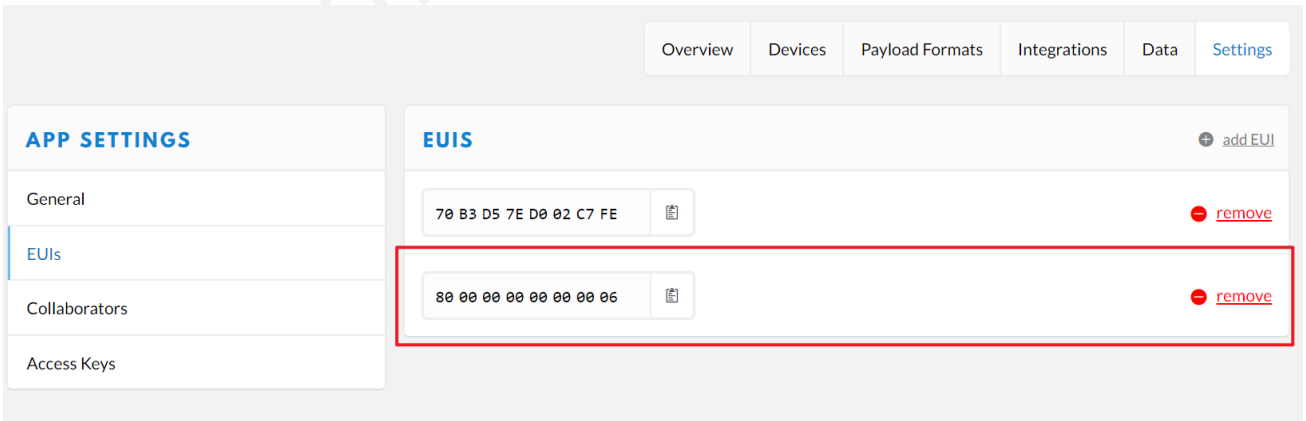
(3)





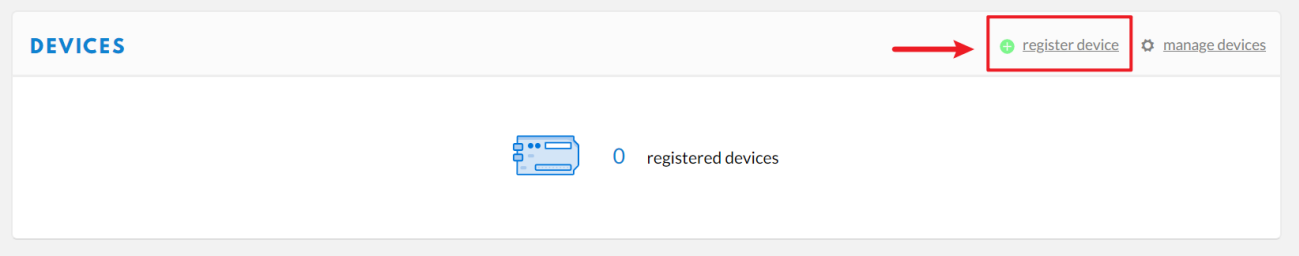


① Application → Application EUIS → Manage EUIs.
② →Add EUI.
③ Enter the node's AppEui that you got in the 3.1 step.
④ →Add EUI.

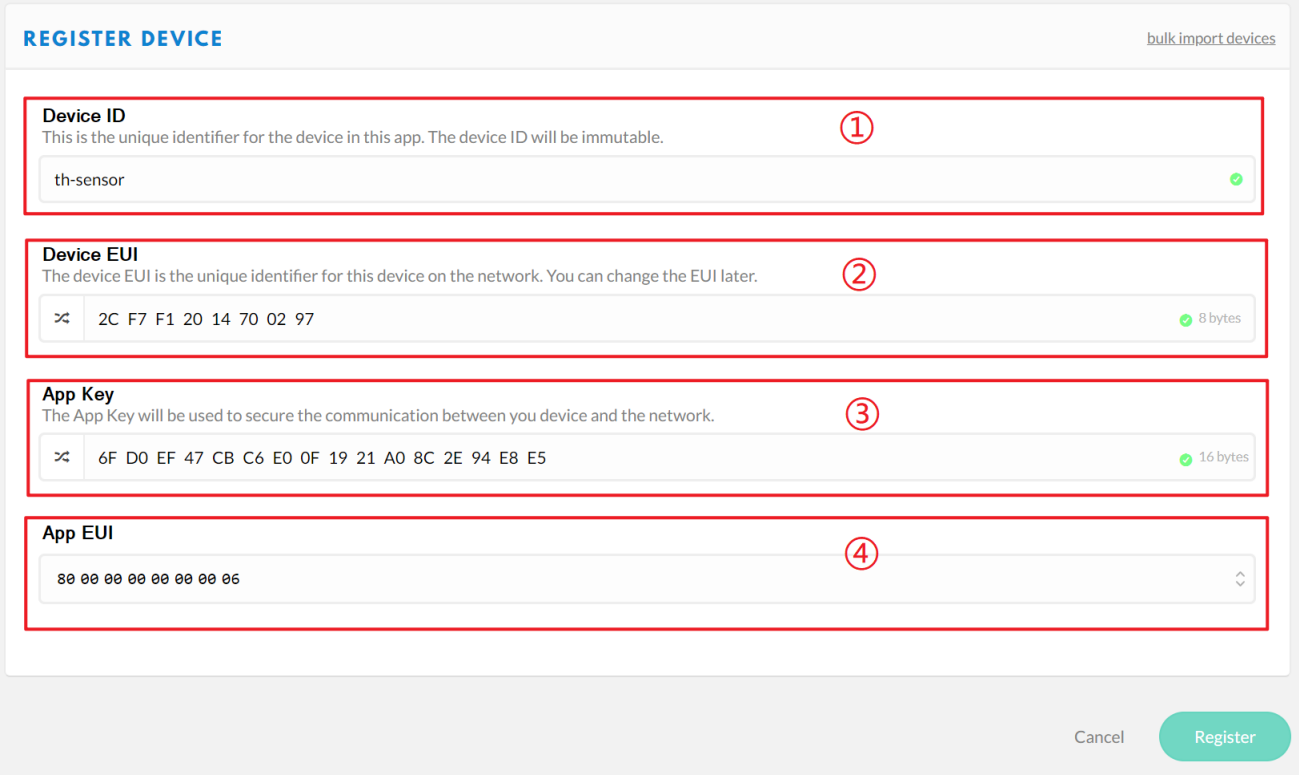


4.3 Sensor Node Registration on TTN

(1) Application → Devices → register device



(2)



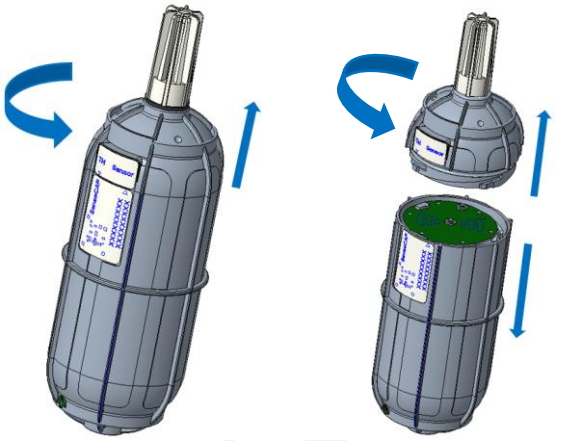
① Device ID: Enter a unique name.
② Device EUI: Enter the node's Device EUI that you got in the 3.1 step.
③ App Key: Enter the node's App Key that you got in the previous step.
④ App EUI: Select the node's App EUI.
⑤ Register.

4.4 Connect the Node to TTN

4.4.1 Power on

The power switch is hidden inside the device. Open the device and turn on the power before installing the sensors. Here is the step-by-step instruction:

1) Loosen the Sensor Probe by turning the cap counterclockwise. Use the white cap opener to make this process easier. The image below uses TH Sensor as an example and applies to all other SenseCAP sensors.

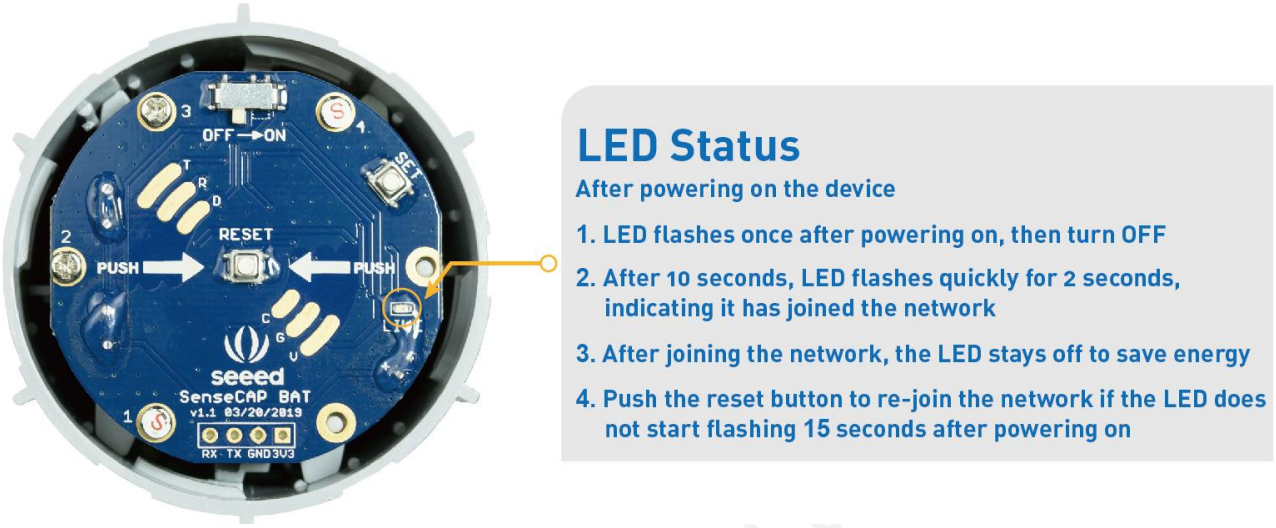


2) After opening the device, turn the switch to “ON”, and the LED on the lower right corner will flash, indicating that the power is on. Wait for about 10 seconds, then the LED will flash quickly for 2 seconds, indicating that the device is connected to the network.

3) After the device is connected to the network, connect the Sensor Probe back with the Sensor Node Controller by turning it clockwise. Please note that the labels on both parts should be aligned as shown in the image below, otherwise the two parts will not be attached to function properly and data will not be uploaded.

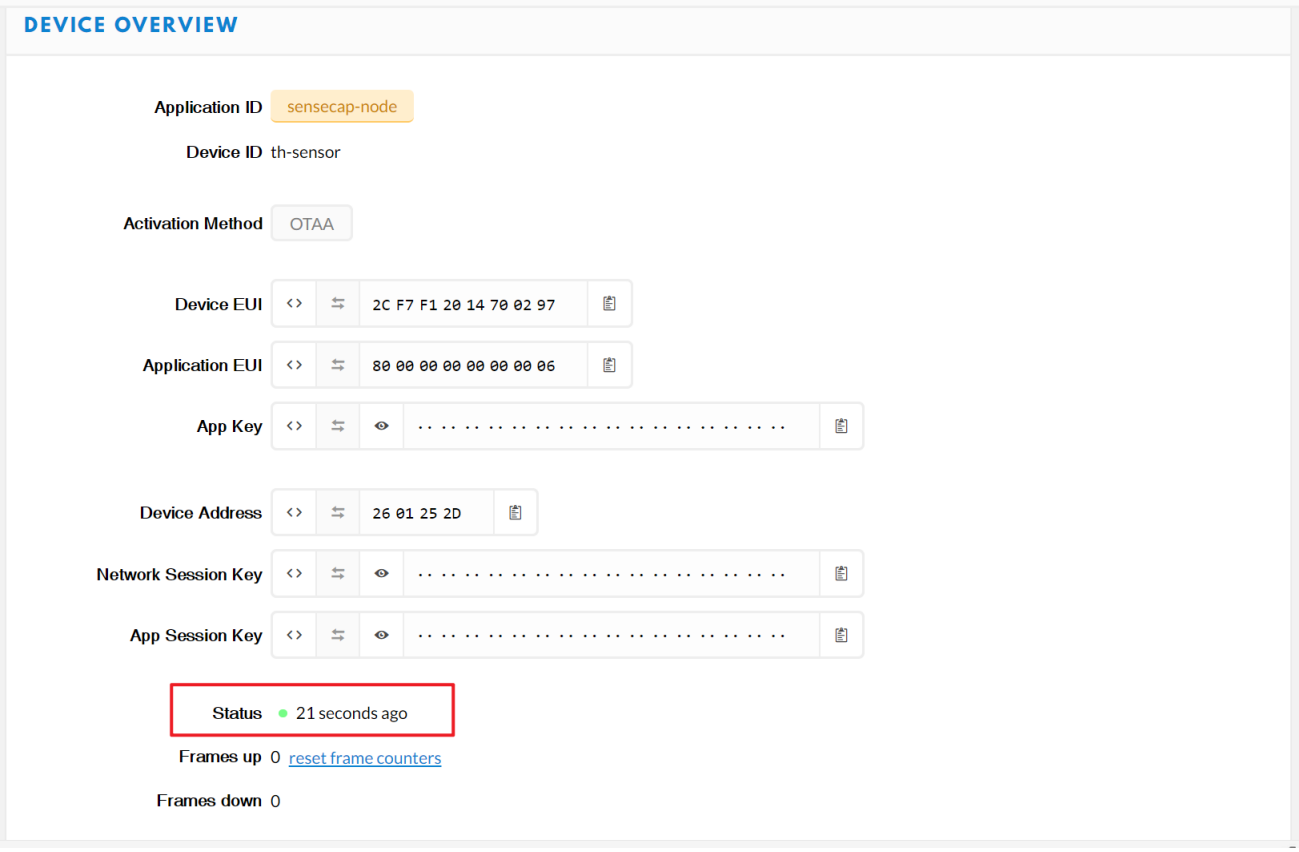
4.4.2 Sensor Node Working Status

You can refer to the LED indicator for the Sensor Node for its working status. Please see the status explanations in the image below:

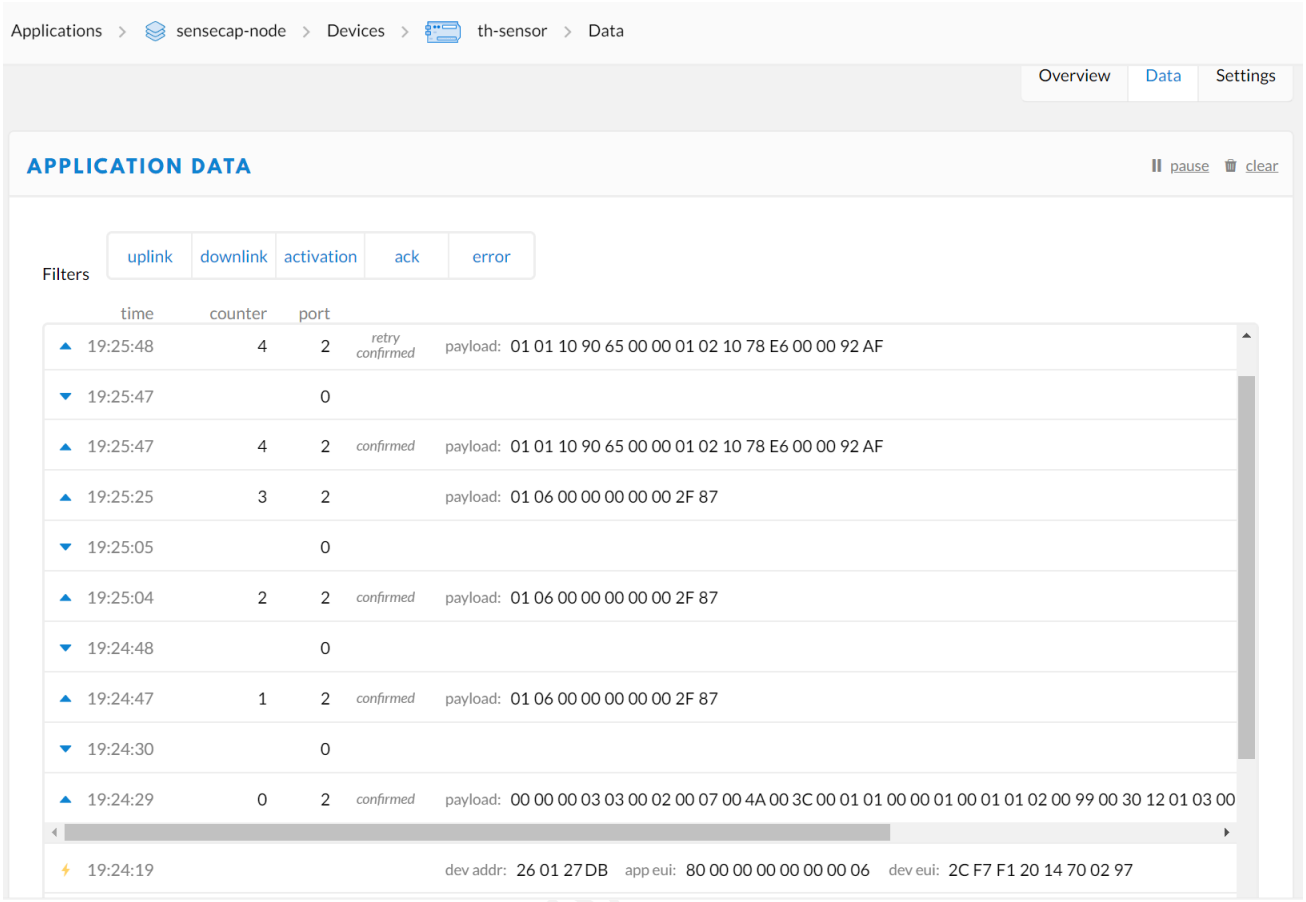


4.4.3 Checking Sensor Node Connection to the TTN

(1) On the Device Overview page, Status turns green.



(2) On the Data page, data package is uploaded. For the format of the payload, refer to the Decoding section.



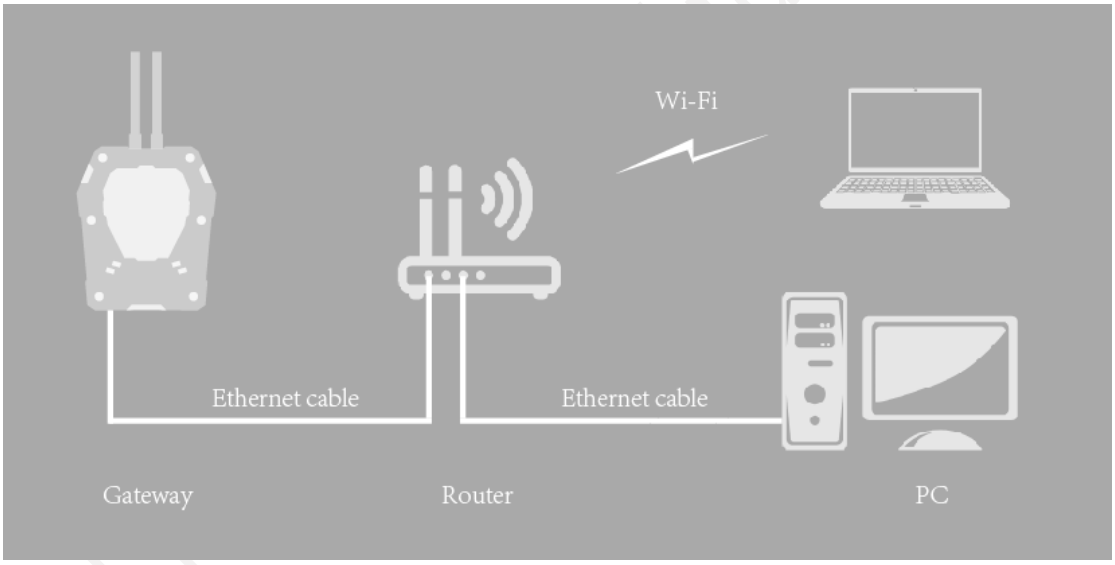
5 Add Gateway to ChirpStack LoRaWAN Network Server Stack

ChirpStack provides open-source components for LoRaWAN networks. Together they form a ready-to-use solution including an user-friendly web-interface for device management and APIs for integration.

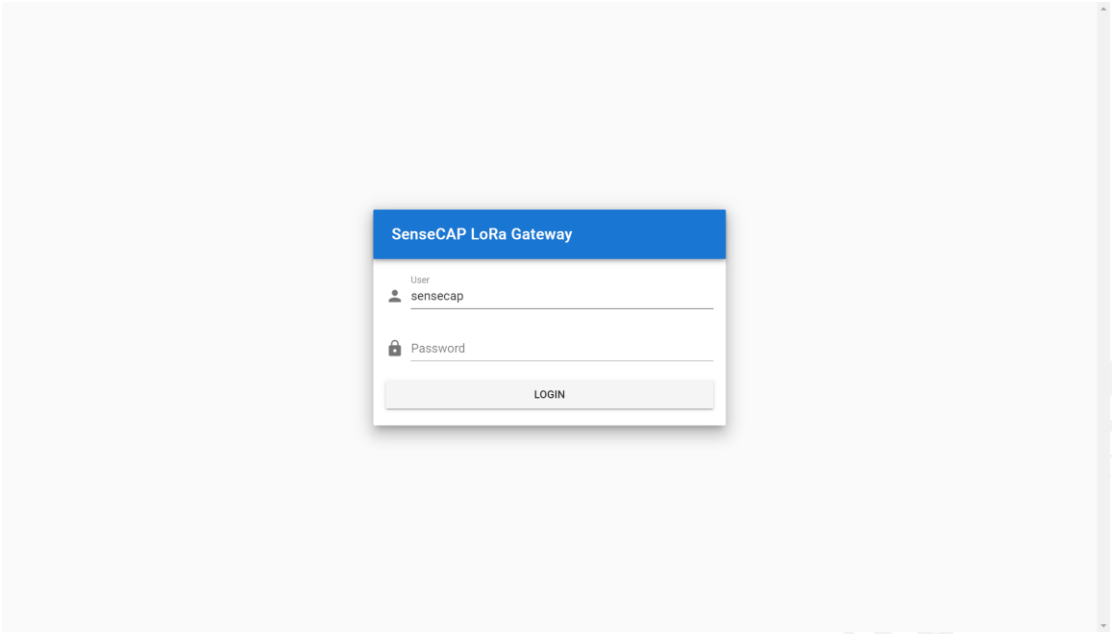
SenseCAP LoRaWAN Gateway has already integrated with ChirpStack LoRaWAN Network Server stack (hereinafter called the "ChirpStack LoRa Server"). The following LoRa Server components are accessible and configurable in Gateway: ChirpStack Gateway Bridge, ChirpStack Network Server and ChirpStack Application Server.

5.1 Turn on ChirpStack LoRa Server Mode

Prepare a router, and the network connection is shown in the figure:

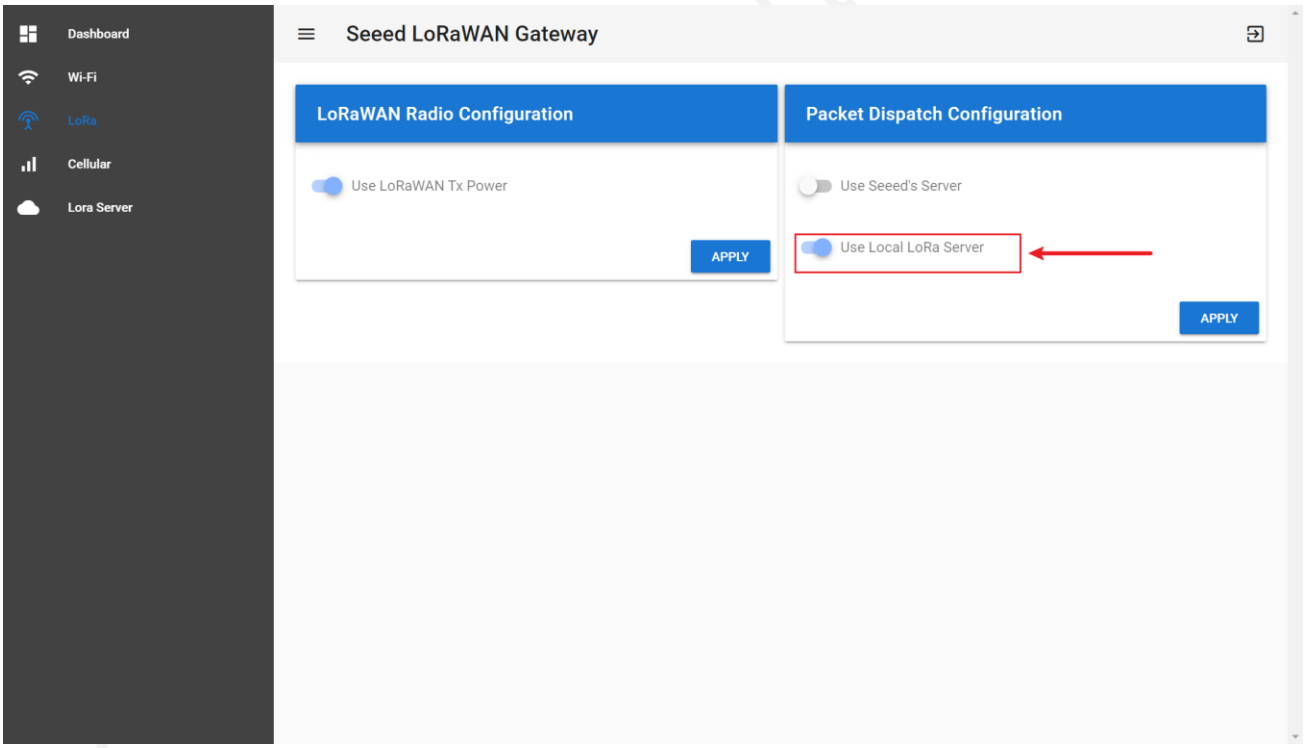


(1) Check the IP of "sensecap" in the background of the router.
(2) Enter IP in the browser: IP:8000
If the IP is 192.168.1.1, enter 192.168.1.1:8000

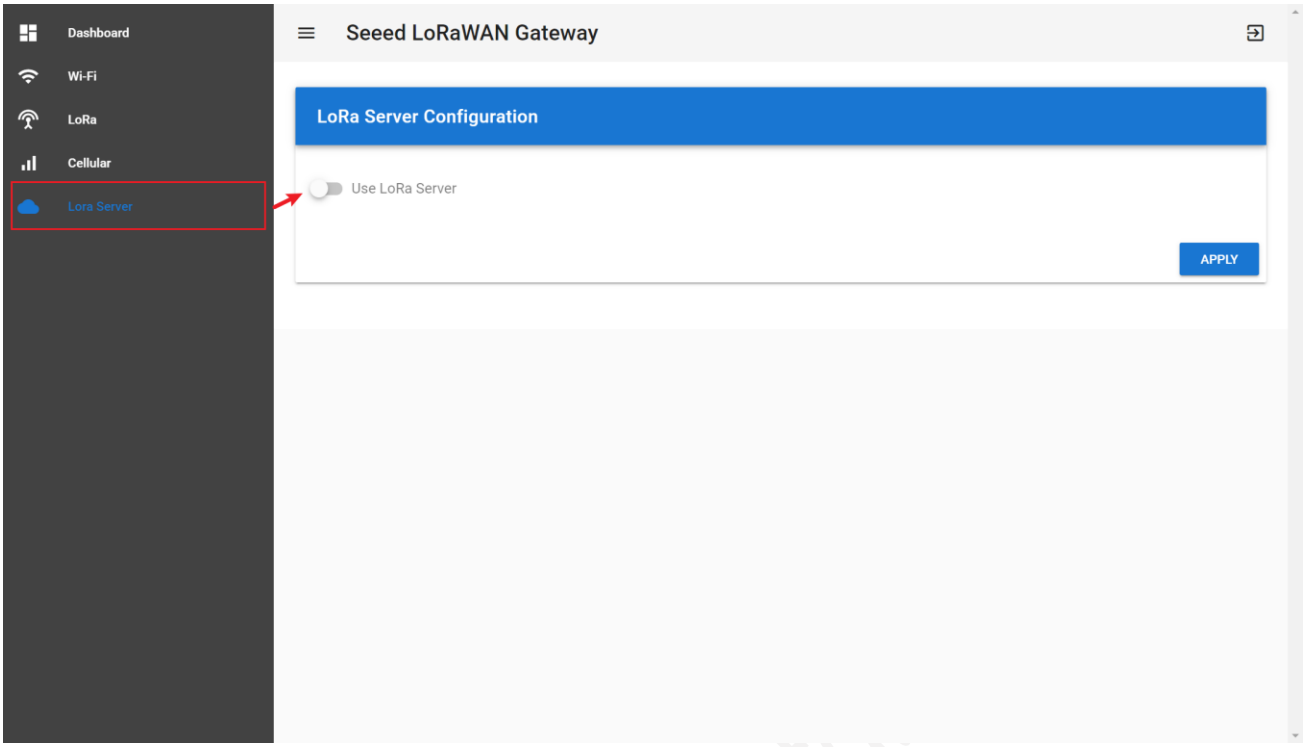


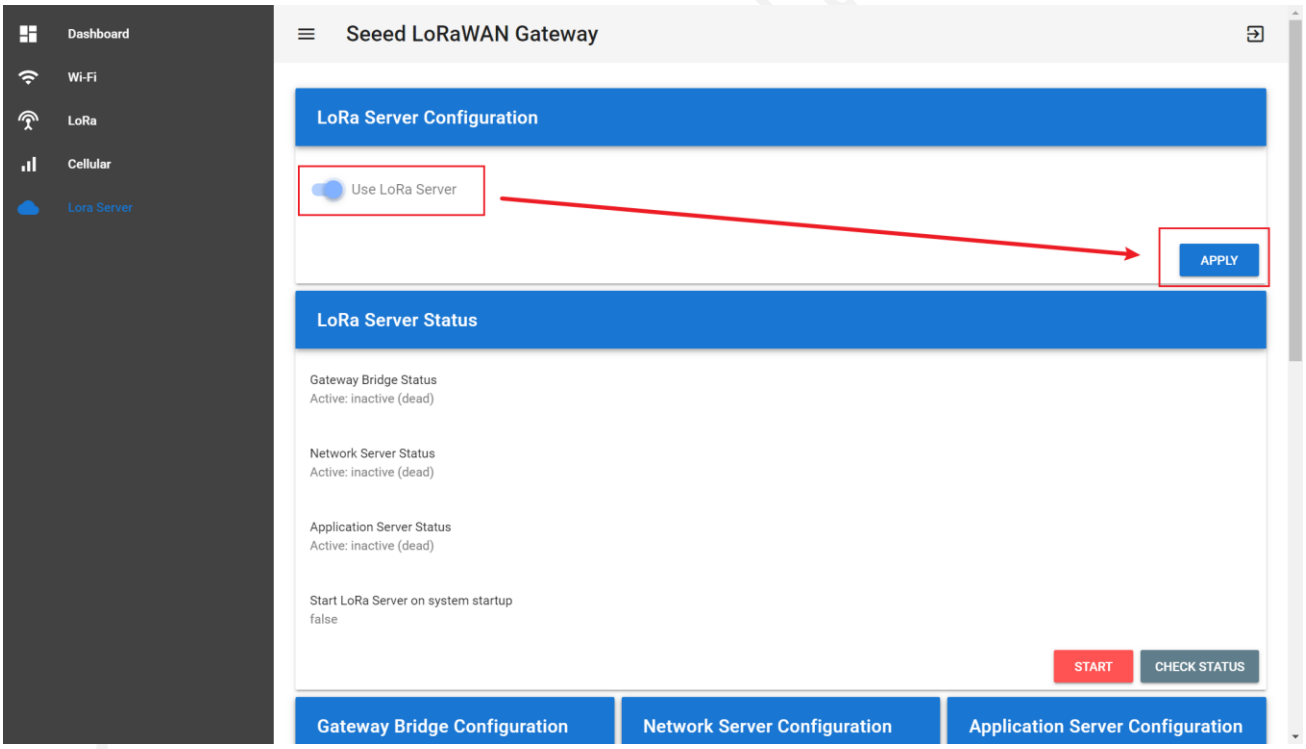
(3) User: sensecap
Password: sensecap!!!

(4) Turn off the “Use Seeed’s Server”, and turn on “Use Local LoRa Server”.



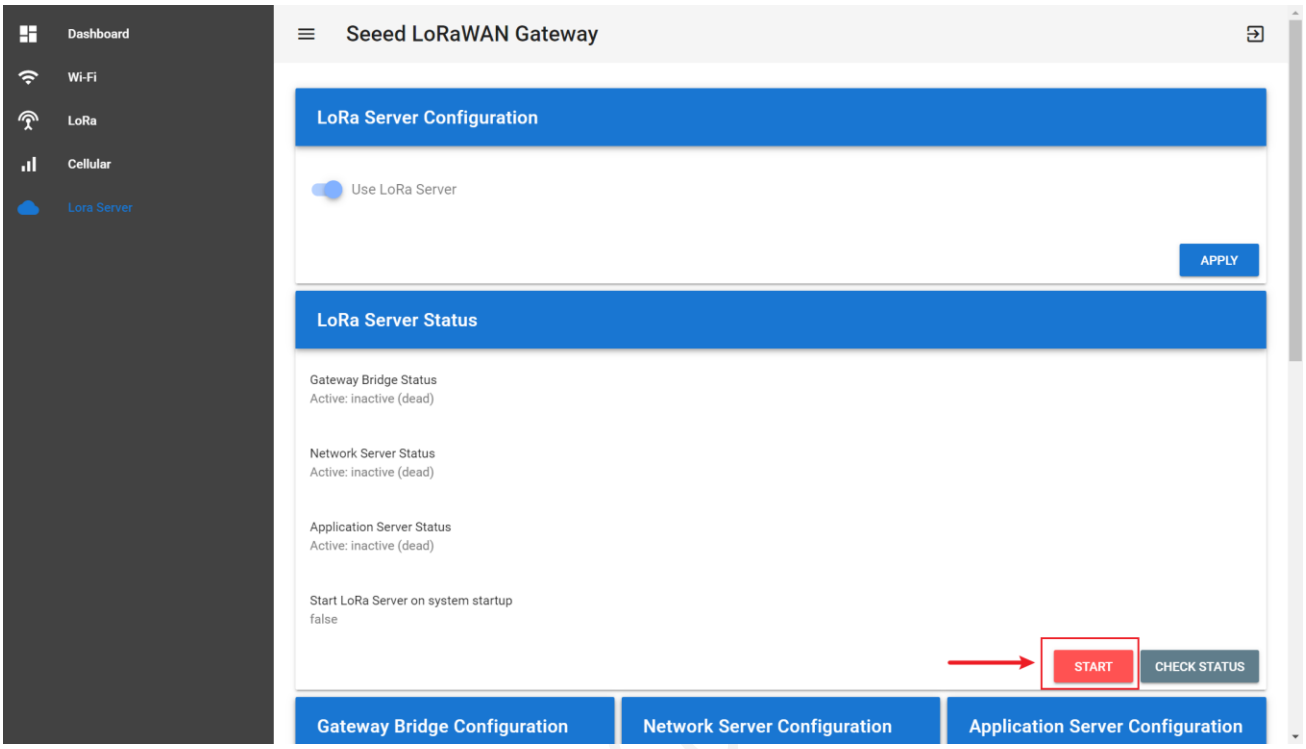
(5) Turn on the “Use LoRa Server” button, and apply. (“LoRa Server” is the name of ChirpStack LoRa Server)





5.2 ChirpStack LoRa Server Configuration

First, click the “Start” button to start the service.

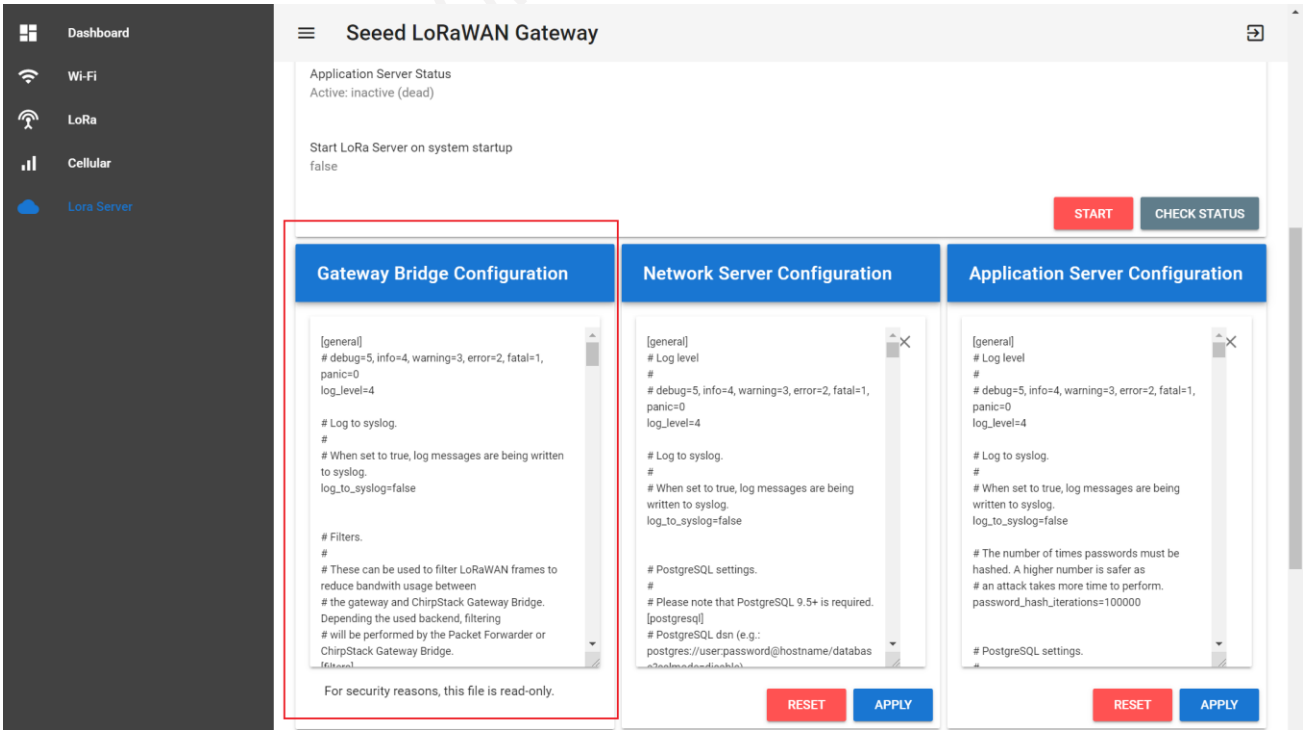


(1) ChirpStack Gateway Bridge:

Refer to: https://www.chirpstack.io/gateway-bridge/

It converts LoRa® Packet Forwarder protocols into a ChirpStack Network Server common data-format (JSON and Protobuf).

For security reasons, this file is read-only.

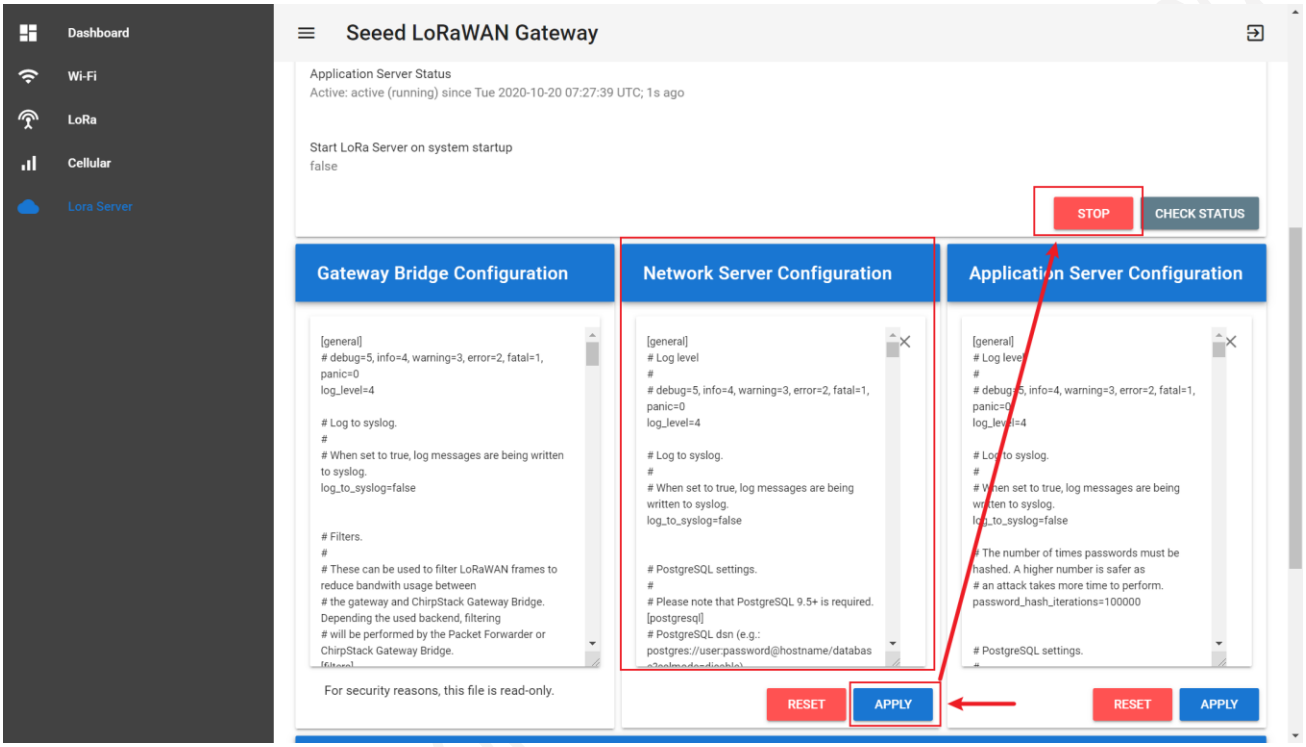


(2) ChirpStack Network Server:

Refer to: https://www.chirpstack.io/network-server/

The responsibility of the Network Server component is the de-duplication of received LoRaWAN frames by the LoRa® gateways and for the collected frames handle the: Authentication; LoRaWAN mac-layer (and mac-commands); Communication with the ChirpStack Application Server; Scheduling of downlink frames.

In general, the default configuration is used. Please refer to the official tutorial before making any modifications.
Click "APPLY" to save the configuration after making changes.
Then, click "STOP" in "Application Server Status" and finally click "START" to make the configuration take effect.

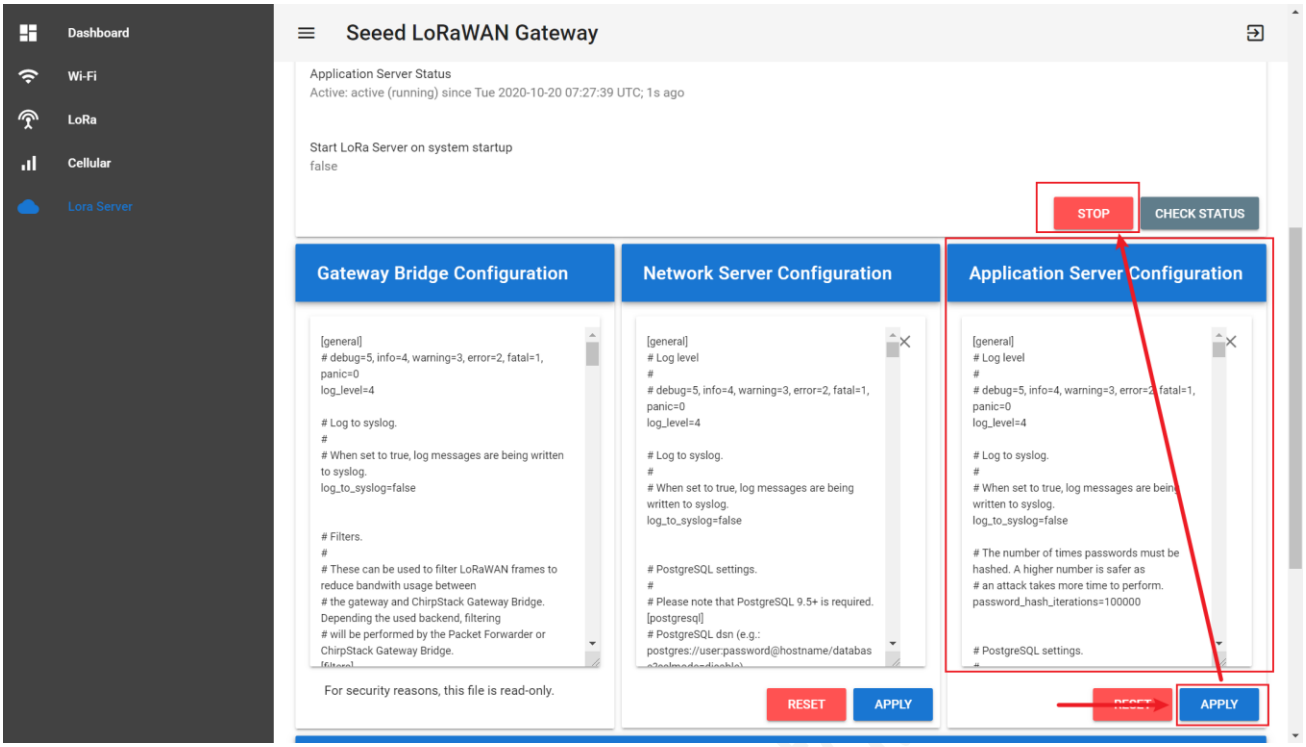


(3) ChirpStack Application Server:

Refer to: https://www.chirpstack.io/application-server/

It is responsible for the device “inventory” part of a LoRaWAN infrastructure, handling of join-request and the handling and encryption of application payloads.

In general, the default configuration is used. Please refer to the official tutorial before making any modifications.
Click "APPLY" to save the configuration after making changes.
Then, click "STOP" in "Application Server Status" and finally click "START" to make the configuration take effect.



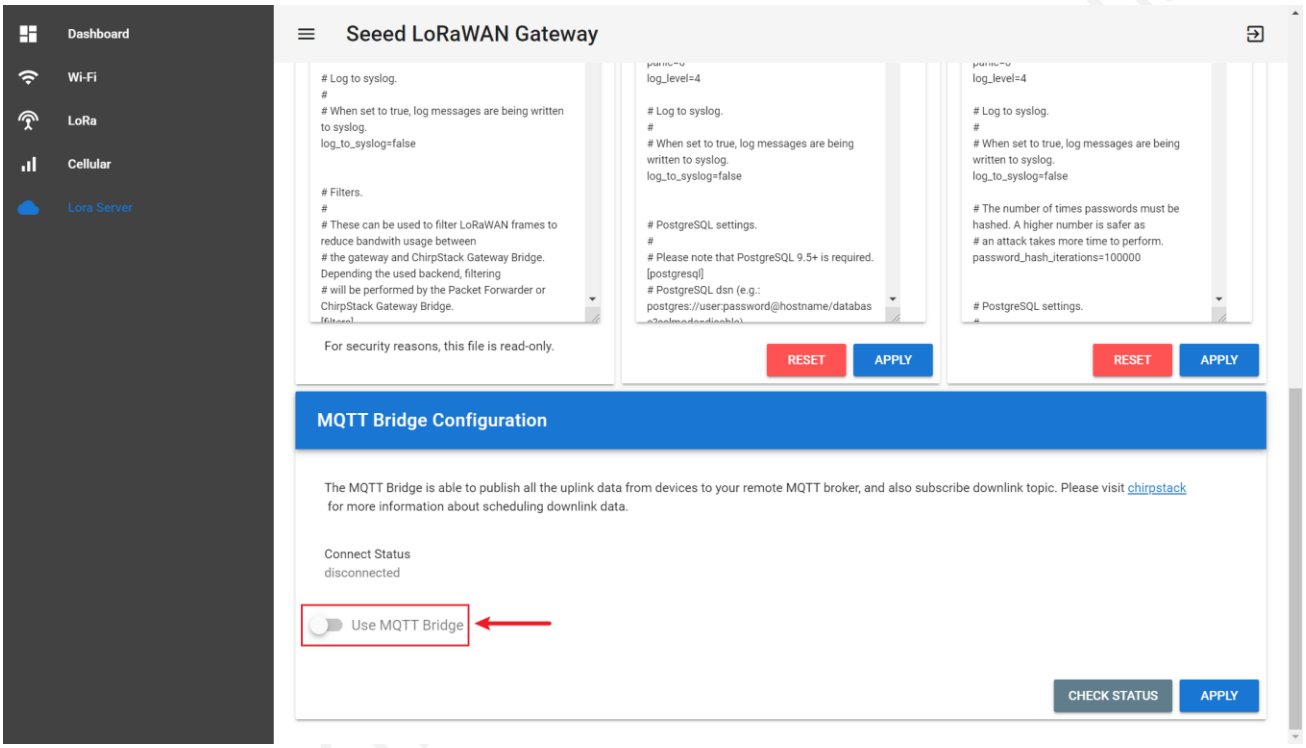
(4) If you have the wrong configuration, click "RESET" to restore the default configuration.

5.3 MQTT Bridge Configuration

The MQTT Bridge is able to publish all the uplink data from devices to your remote MQTT broker, and also subscribe downlink topic. Please visit ChirpStack(https://www.chirpstack.io/application-server/integrations/mqtt/) for more information about scheduling downlink data.

5.3.1 Gateway Configuration

(1) Click “Use MQTT Bridge”.



(2) After filling in each parameter, click "APPLY".

①

MQTT Server address: mqtt://xxx.xx or mqtts://xxx.xx

If xxx.xx (IP) is 111.230.200.102, the address is mqtt://111.230.200.102 or mqtts://111.230.200.102
If xxx.xx (url) is mybroker.com, the address is mqtt:// mybroker.com or mqtts:// mybroker.com

②

MQTT Server ‘s Port.
In general, mqtt corresponds to port 1883 and mqtts to port 8883.

③

Keepalive:
60 is default value. When the MQTT connection between the Gateway and the Server is disconnected over 60 seconds, it determines that the client is offline.
0 means turn off the keepalive function.

④

CleanSession:
true: the gateway reconnects to the network after a power outage or disconnection, and cannot receive data from MQTTpub to the gateway for that period.
false: the gateway reconnects to the network after a power outage or disconnection, and can receive data from MQTTpub to the gateway for that period.

⑤

Username: Null if none, depending on the server configuration.

⑥

Password: Null if none, depending on the server configuration.

⑦

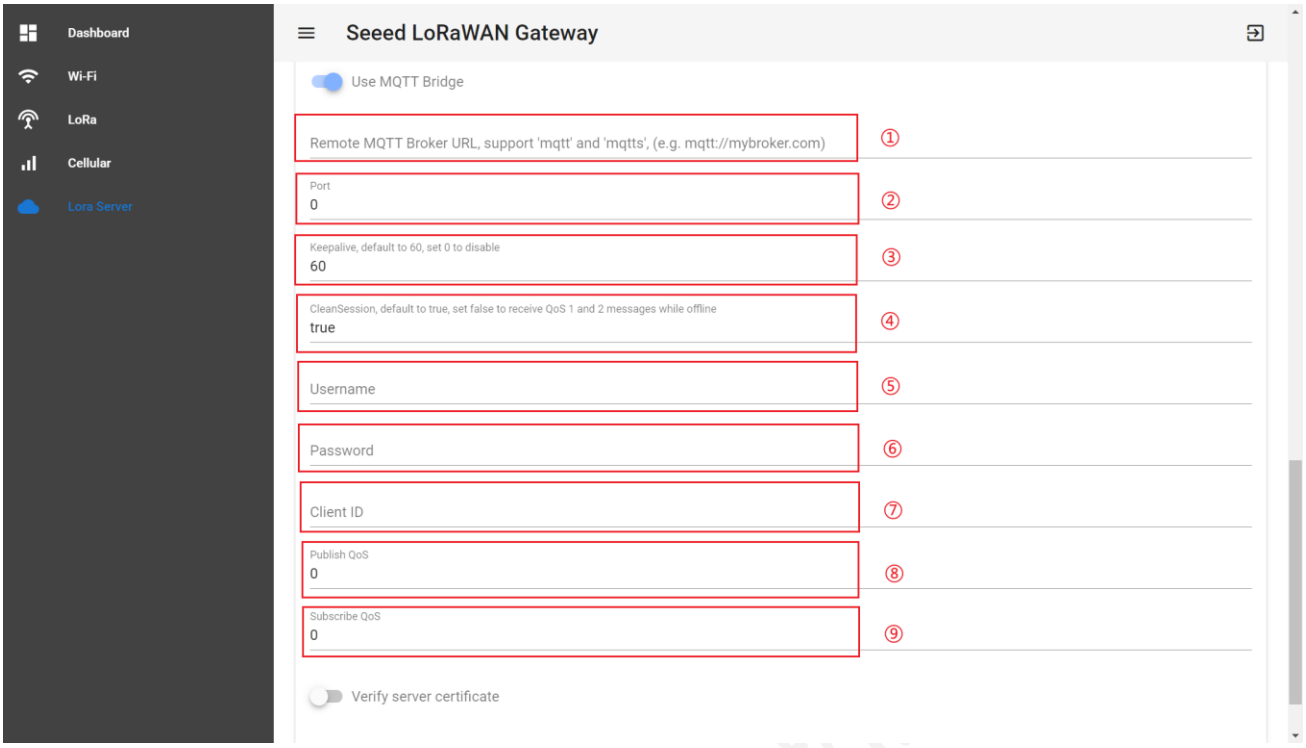
Client ID: Custom the name, and each Client ID is unique to the same MQTT server.

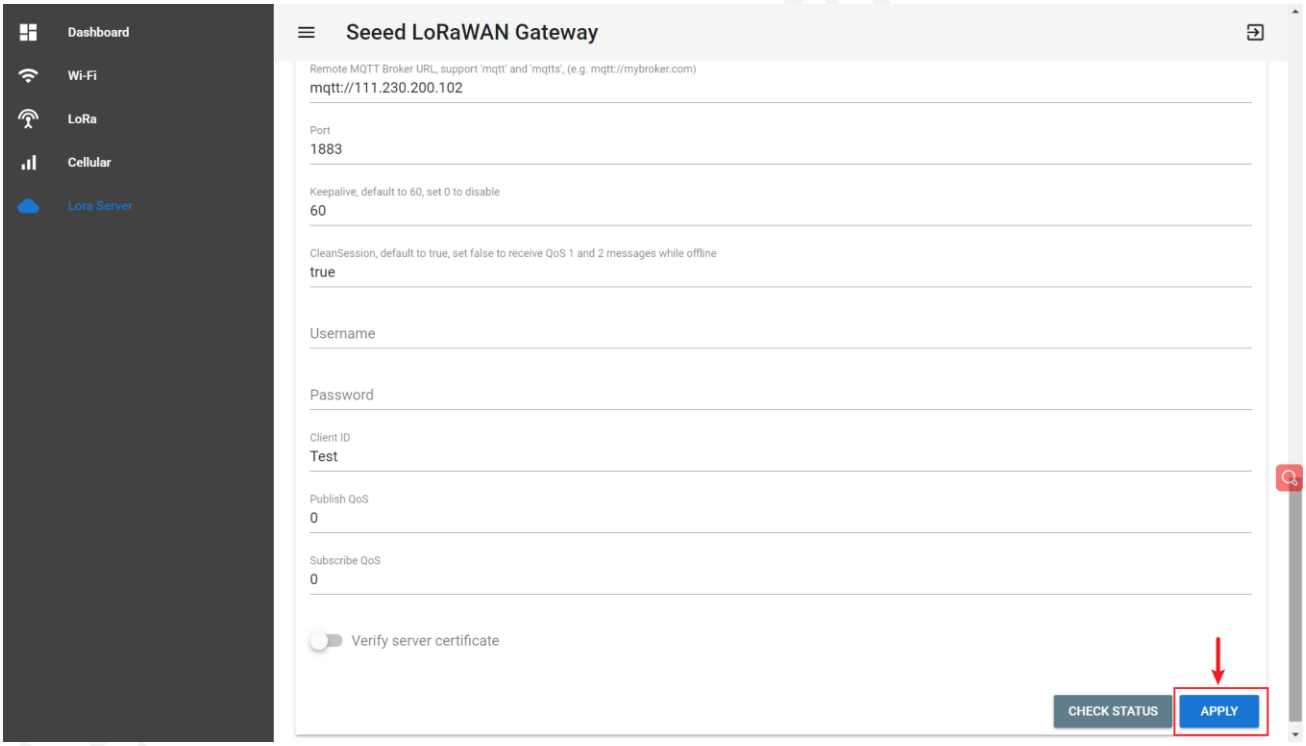
⑧

Publish QoS: 0, 1 or 2. (refer to the MQTT rules)

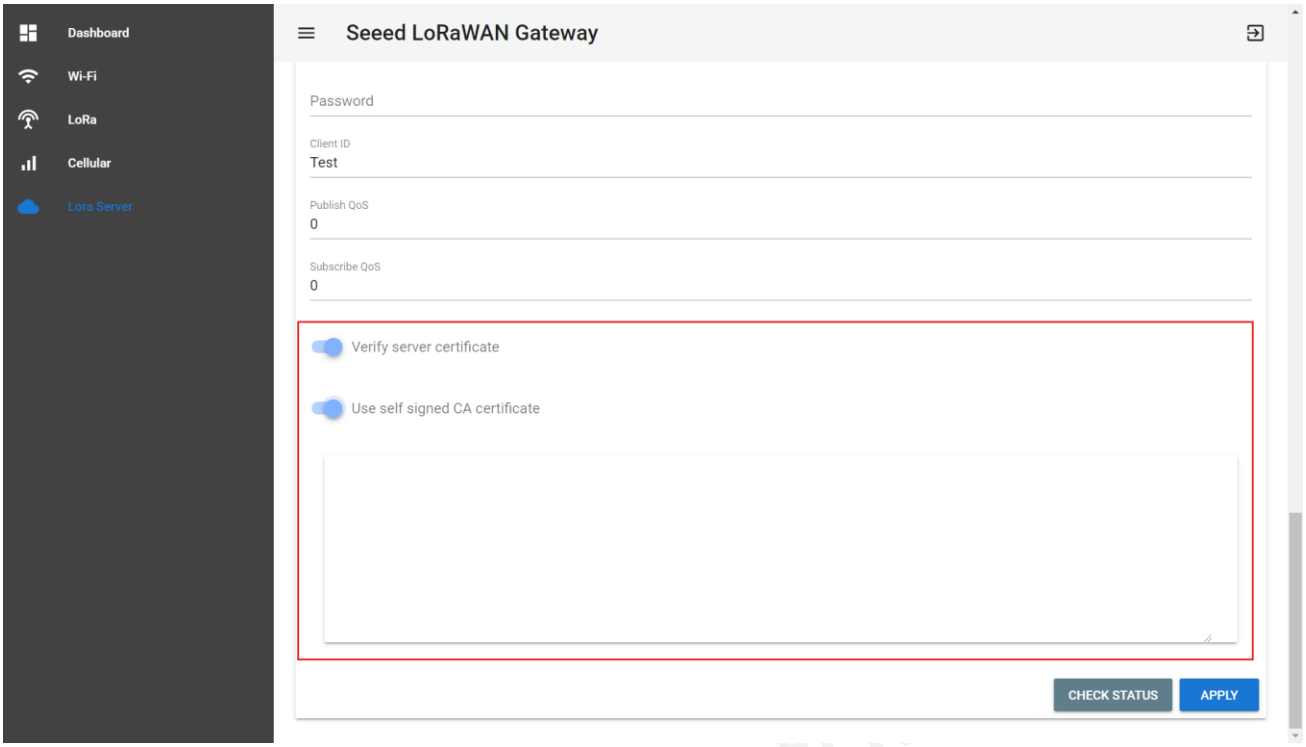
⑨

Subscribe QoS: 0, 1 or 2. (refer to the MQTT rules)

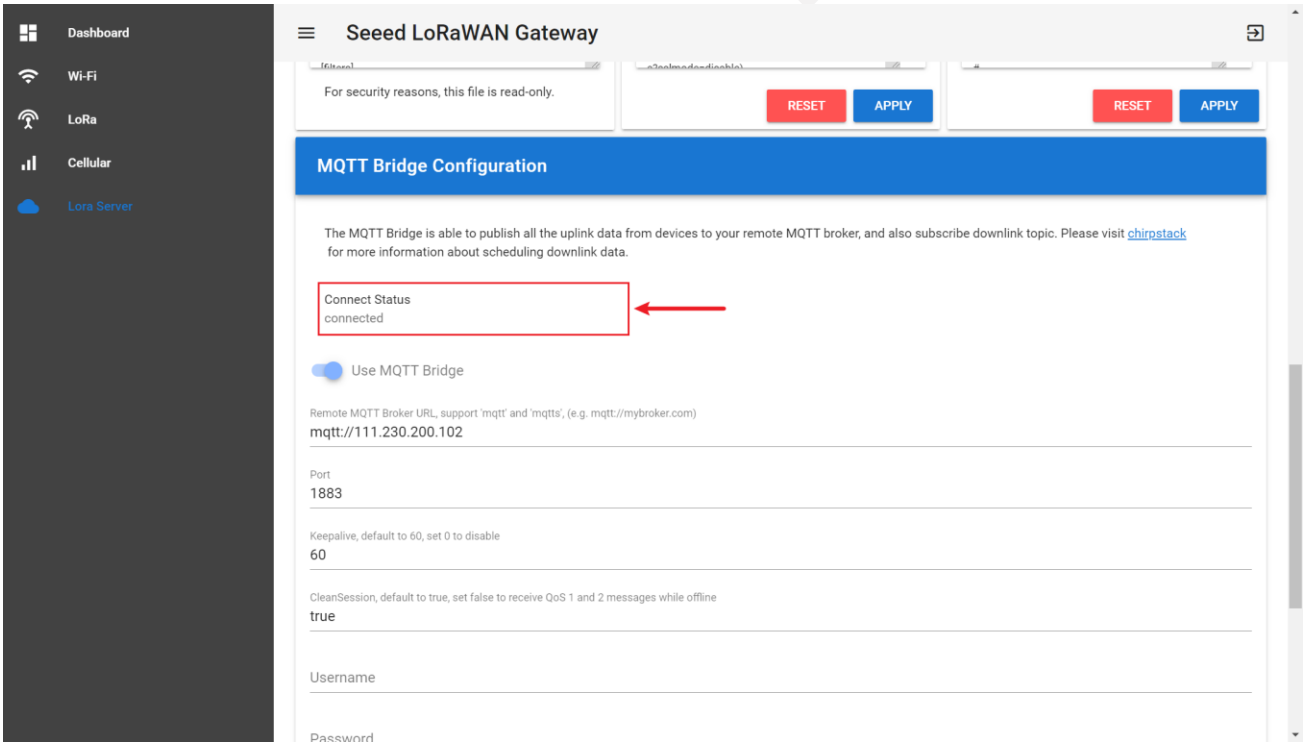




(3) It is off by default and can generally be ignored: Verify server certificate.
If true, the server certificate is verified against the list of supplied CAs.
If false, the server certificate is verified against your self-signed certificate.



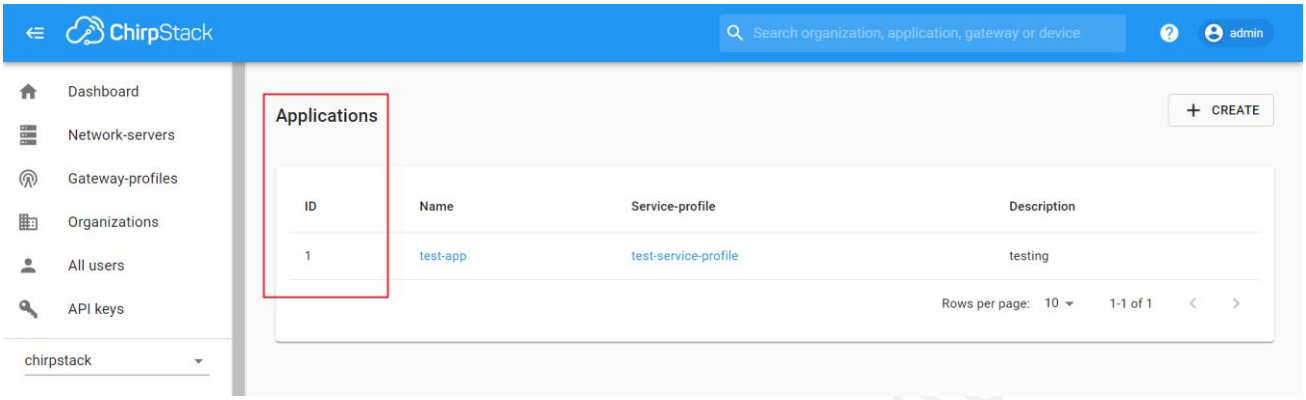
(4) Check Status: Disconnected / Reconnecting / Connected.



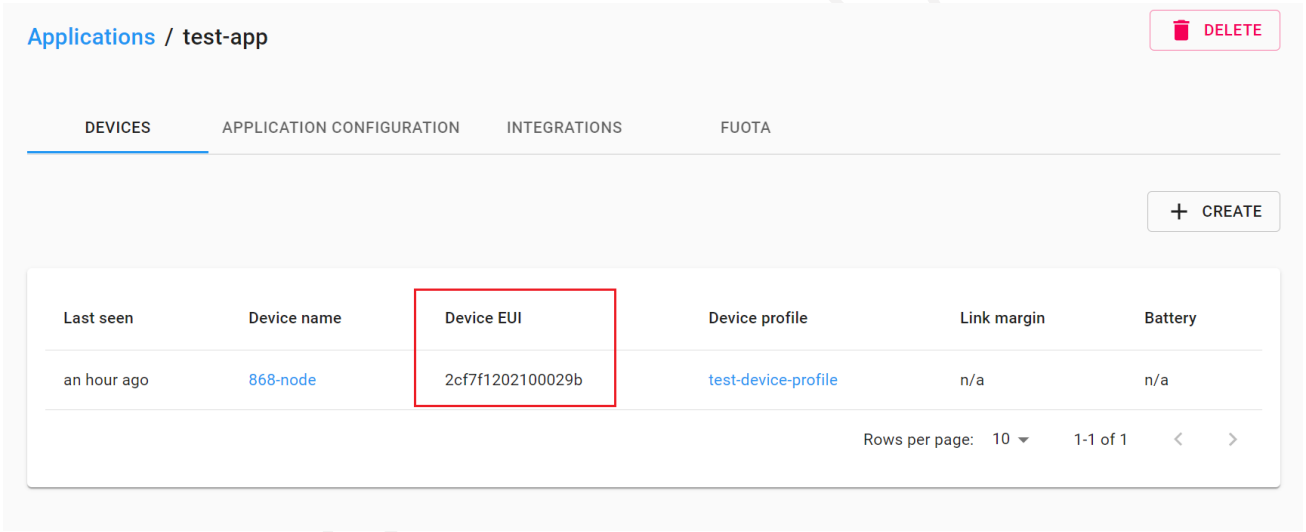
5.3.2 MQTT Client Configuration

For details, please refer to: https://www.chirpstack.io/application-server/integrations/events/#ack

ApplicationID: the Application ID.



DevEUI: Device EUI.



(1) Device data subscription

application/[ApplicationID]/device/[DevEUI]/event/up e.g. application/1/device/ 2cf7f1202100029b/event/up

(2) Join packet subscription

application/[ApplicationID]/device/[DevEUI]/event/join e.g. application/1/device/ 2cf7f1202100029b/event/join

(3) Status packet subscription

application/[ApplicationID]/device/[DevEUI]/event/status e.g. application/1/device/ 2cf7f1202100029b/event/ status

5.3.3 Scheduling a Downlink

The default topic for scheduling downlink payloads is:

```
application/[ApplicationID]/device/[DevEUI]/command/down
```

The ApplicationID and DevEUI of the device will be taken from the topic.

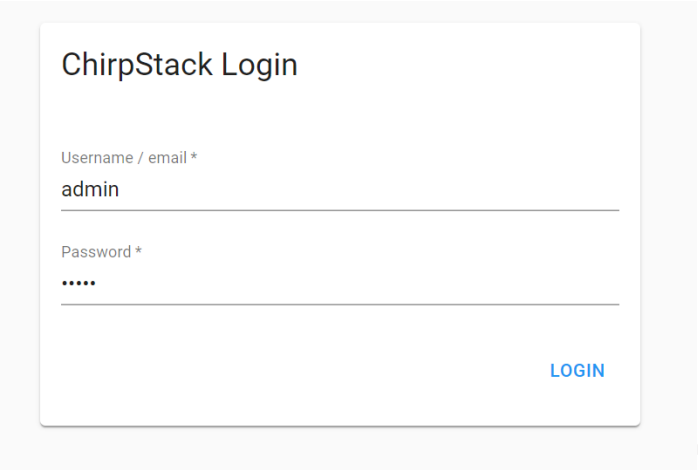
Example payload:

```
{
    "confirmed": true,          // whether the payload must be sent as confirmed data down or not
    "fPort": 10,                // FPort to use (must be > 0)
    "data": "...."              // base64 encoded data (plaintext, will be encrypted by ChirpStack Network Server)
    "object": {                 // decoded object (when application coded has been configured)
        "temperatureSensor": {"1": 25},     // when providing the 'object', you can omit 'data'
        "humiditySensor": {"1": 32}
    }
}
```

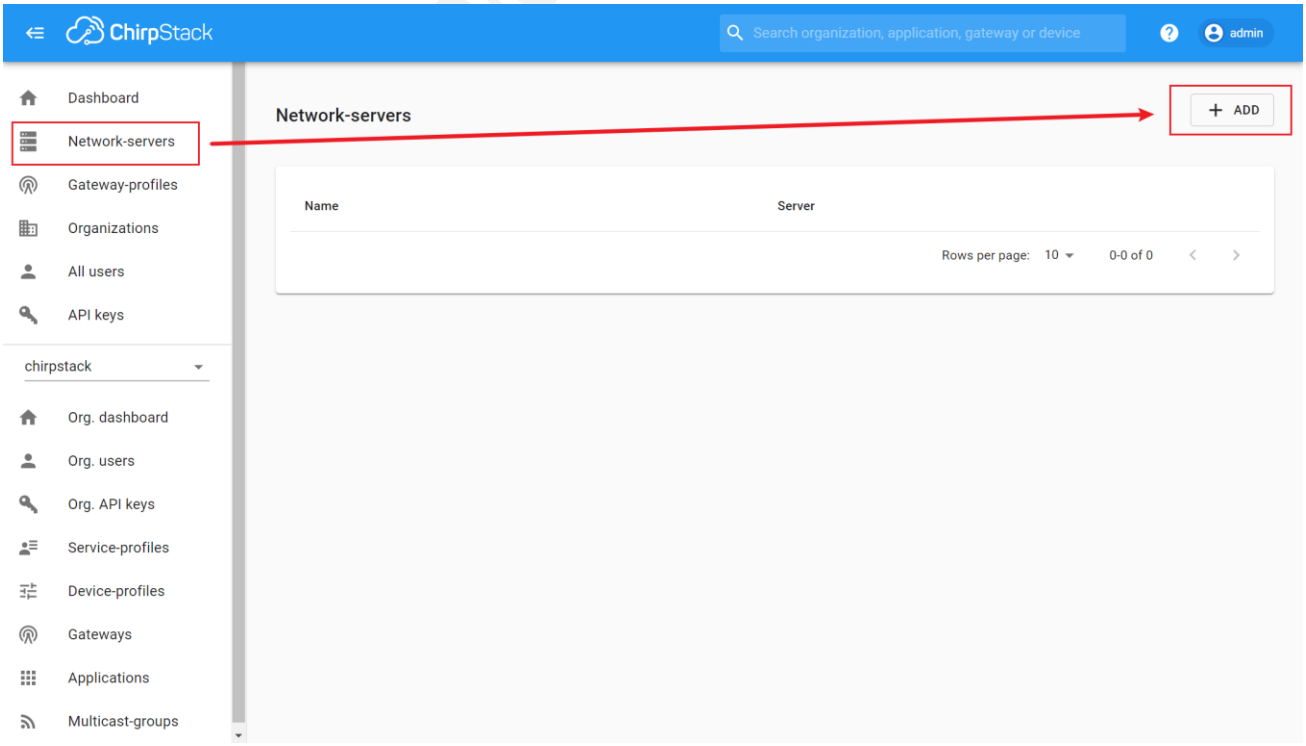
5.4 ChirpStack Application Server

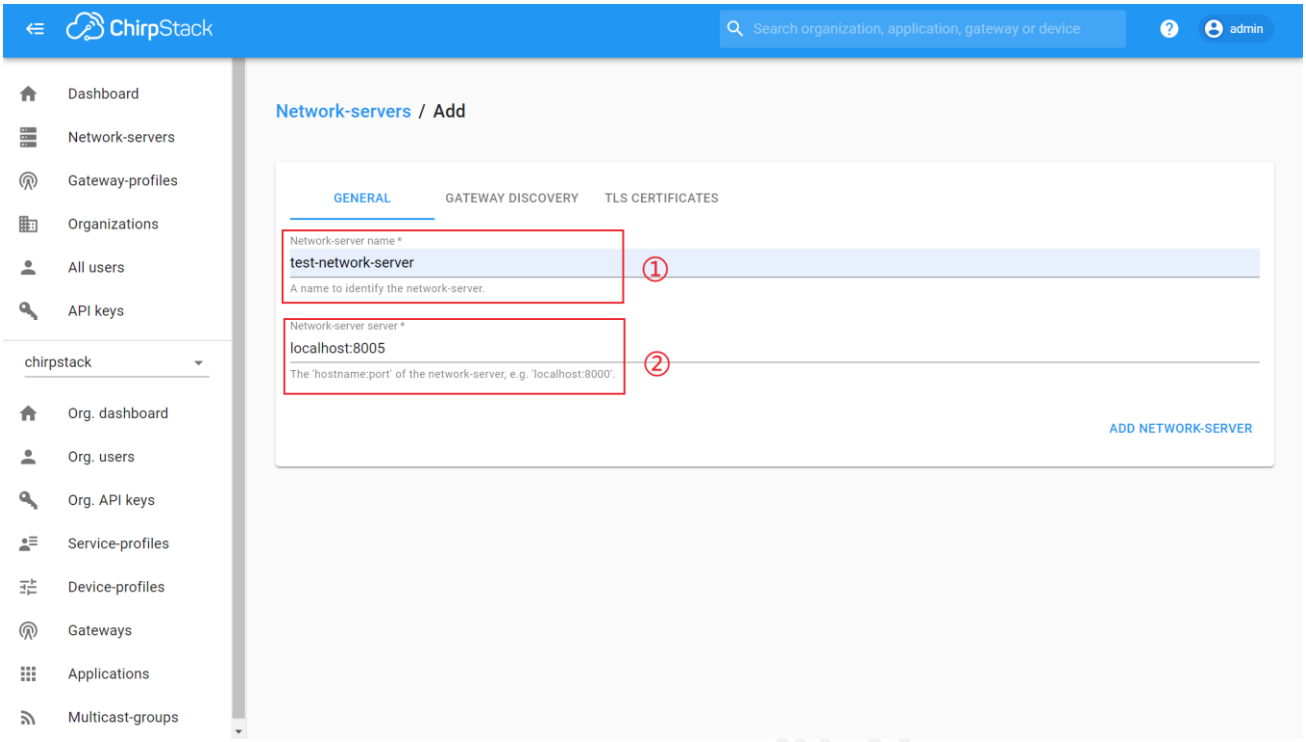
5.4.1 Log on to the background

According to the Gateway IP obtained in Section 4.1, log in the Web UI.
The login address: IP:8080 (if IP is 192.168.8.100, enter 192.168.8.100:8080)
Username(default): admin
Password(default): admin



5.4.2 Add the Network-servers

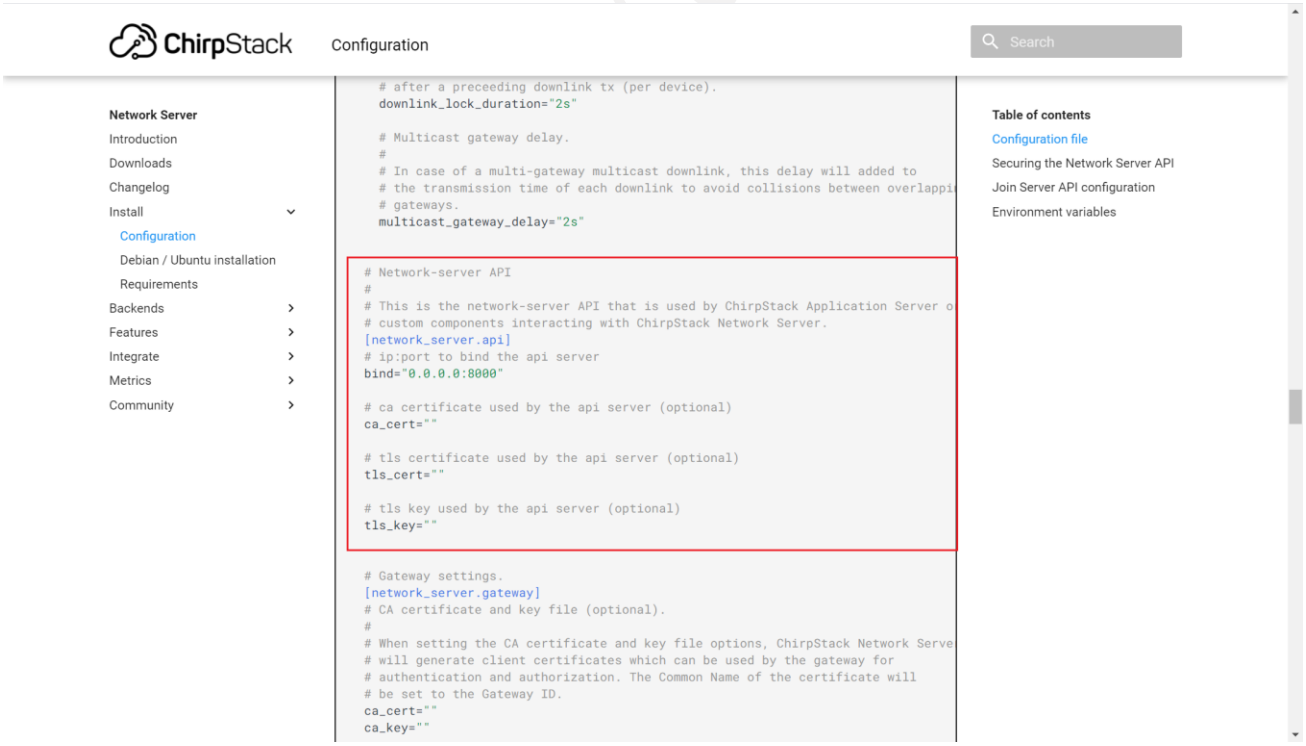


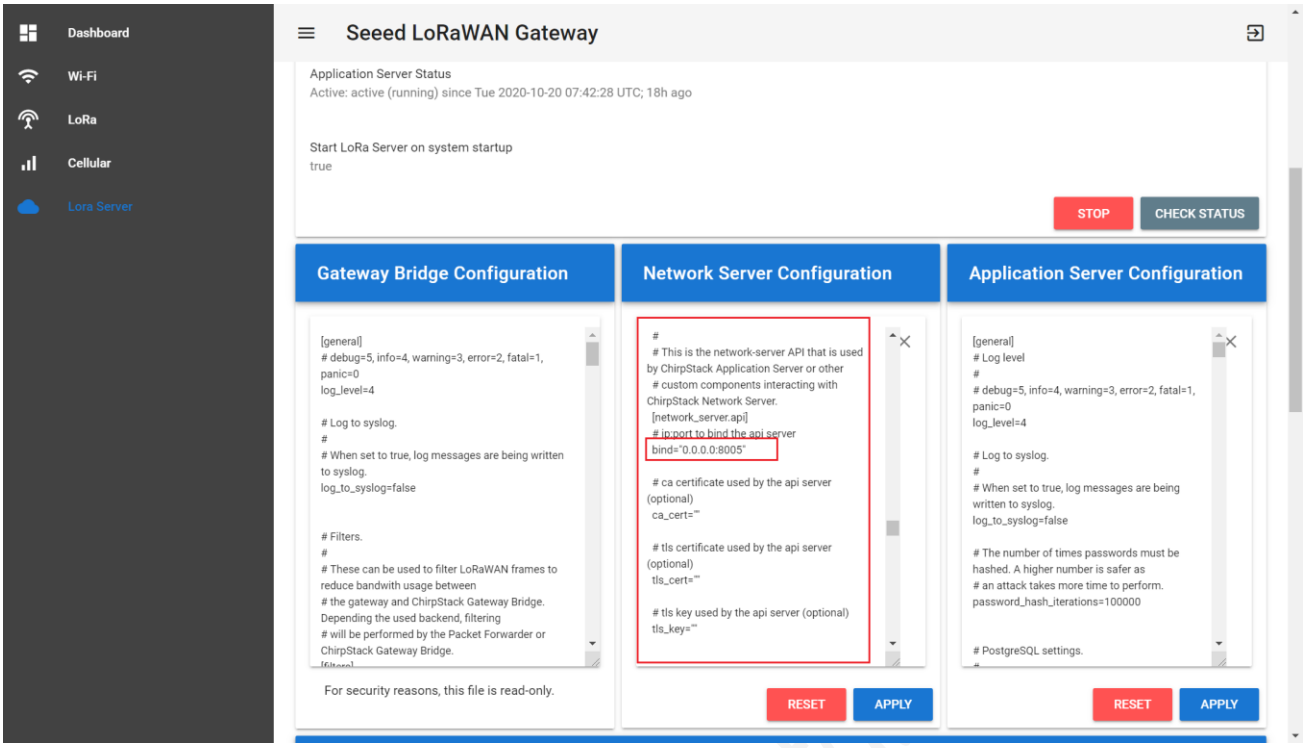


① Network-server name: custom name.

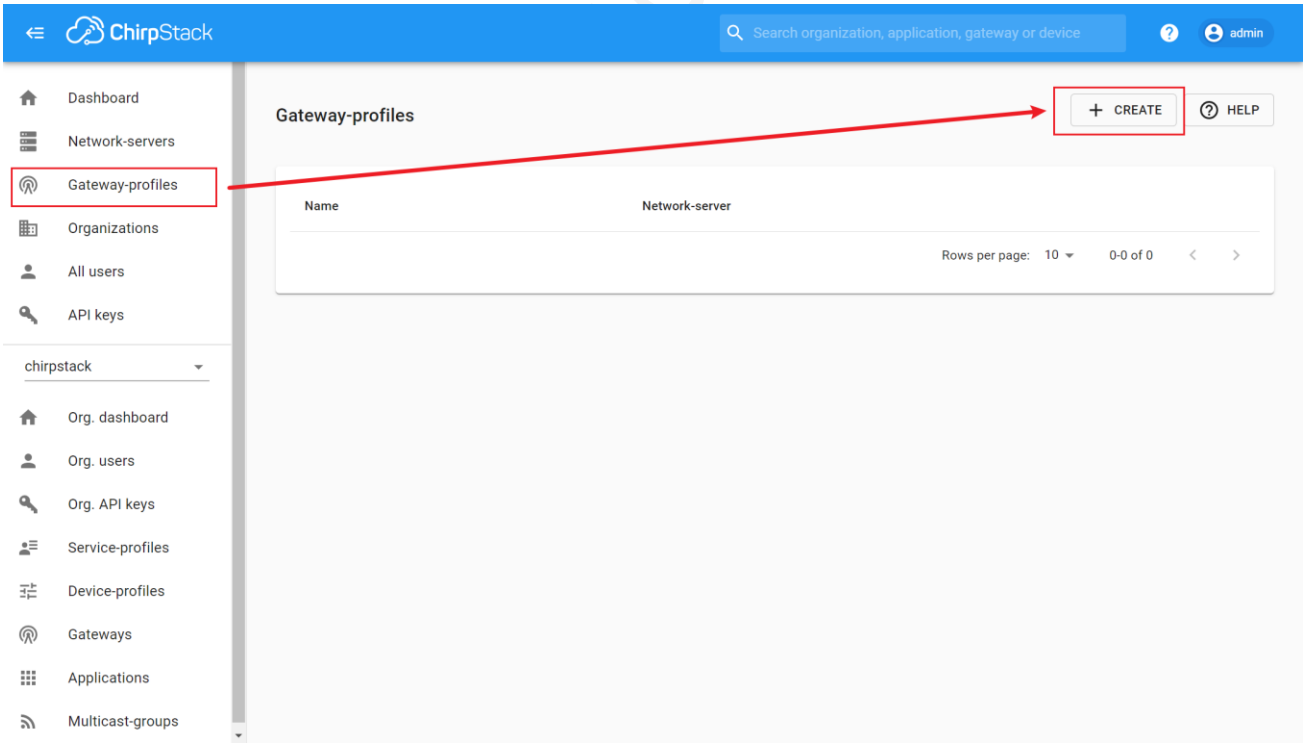
② Network-server server: the default value is localhost:8005

Refer to: https://www.chirpstack.io/network-server/install/config/ . You can modify it in the “Network Server Configuration”.





5.4.3 Create the Gateway-profiles



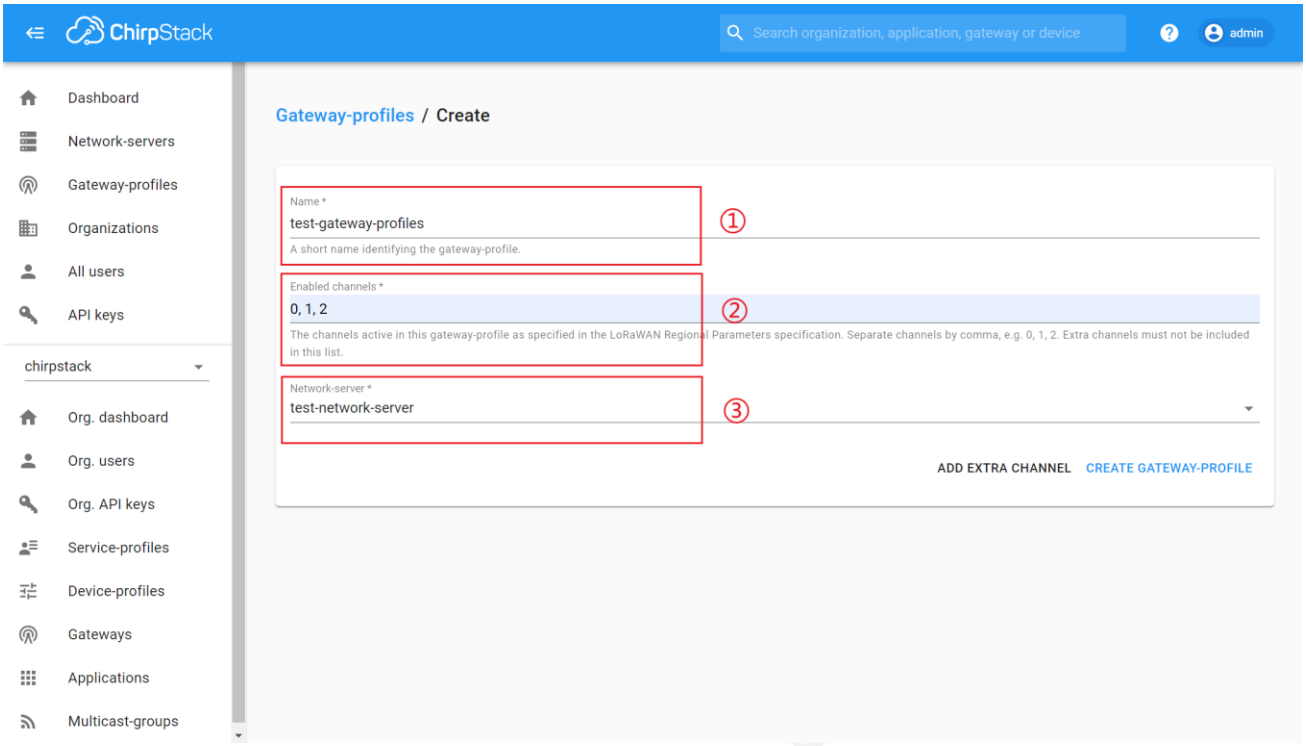
① Name: custom name.

② Enabled channels: 0, 1, 2

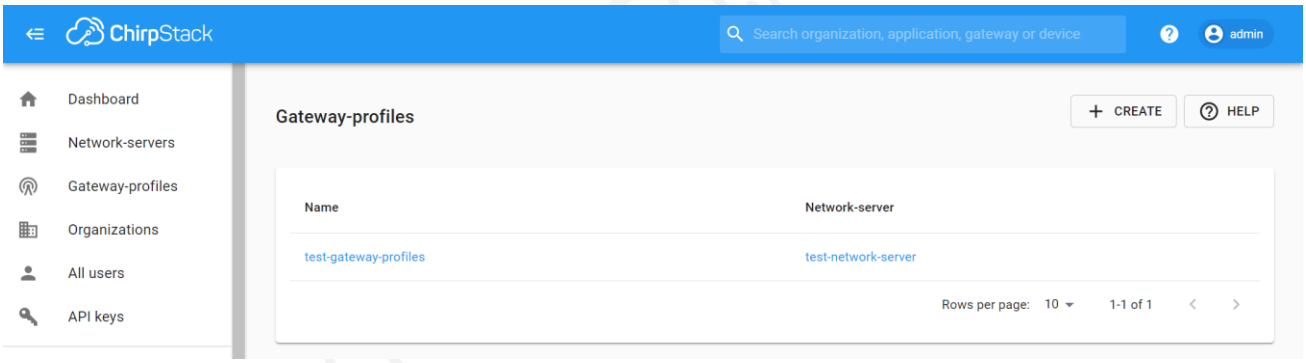
EU channels: 0, 1, 2

US902-923 channels (sub-band 2): 8, 9, 10, 11, 12, 13, 14, 15, 65

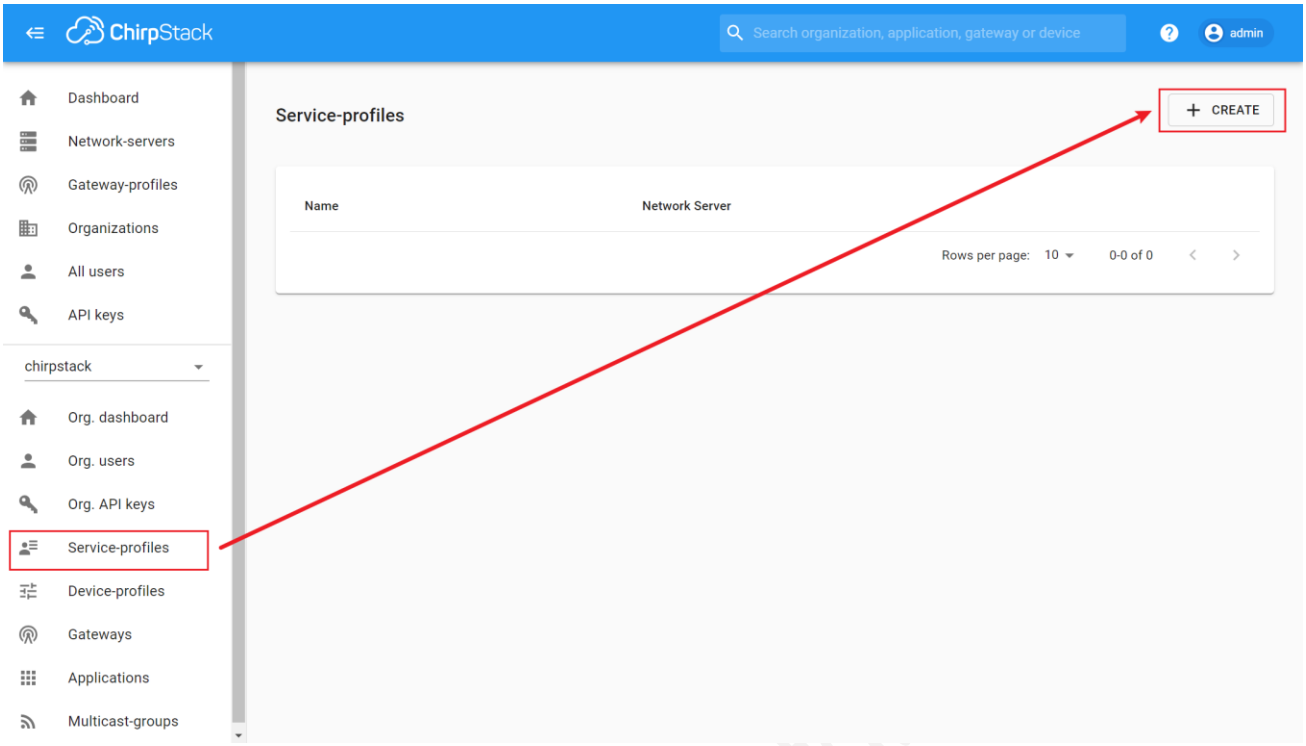
③ Network-server: select the Network-server you created earlier.

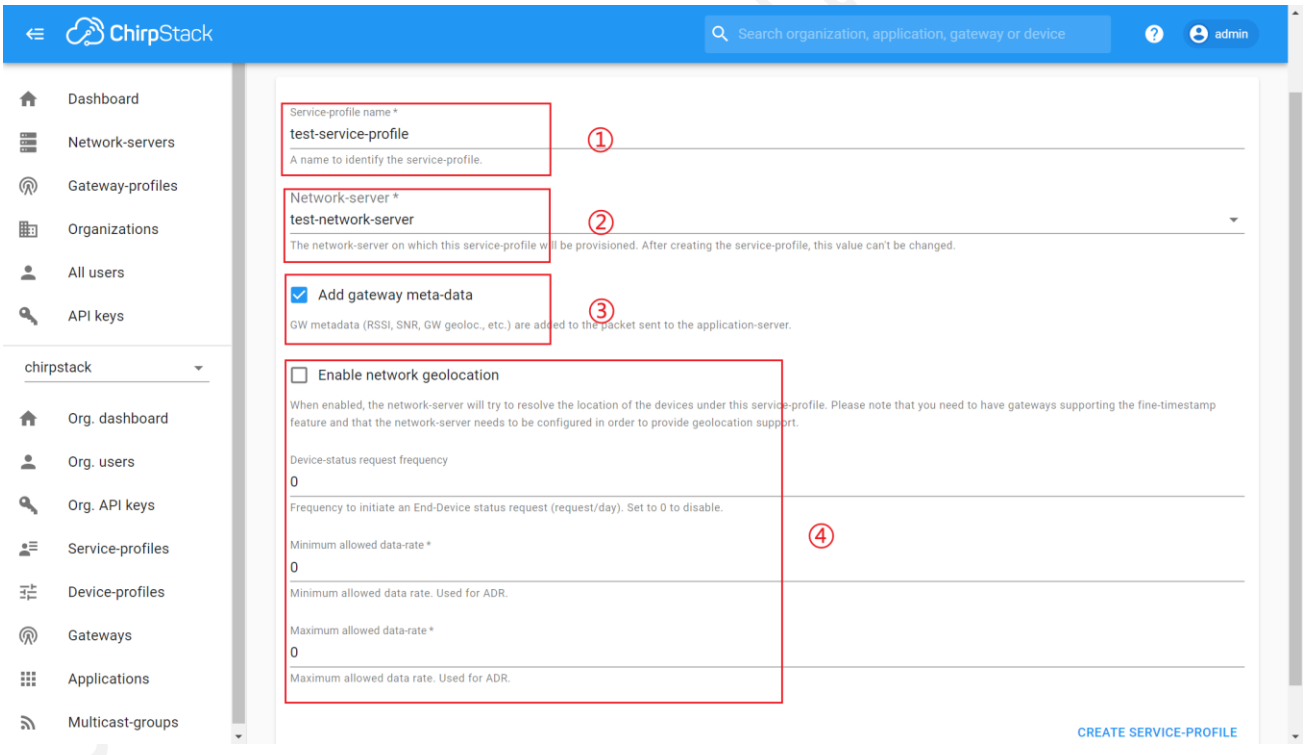


Click the “GREATE GATEWAY-PROFILE”.



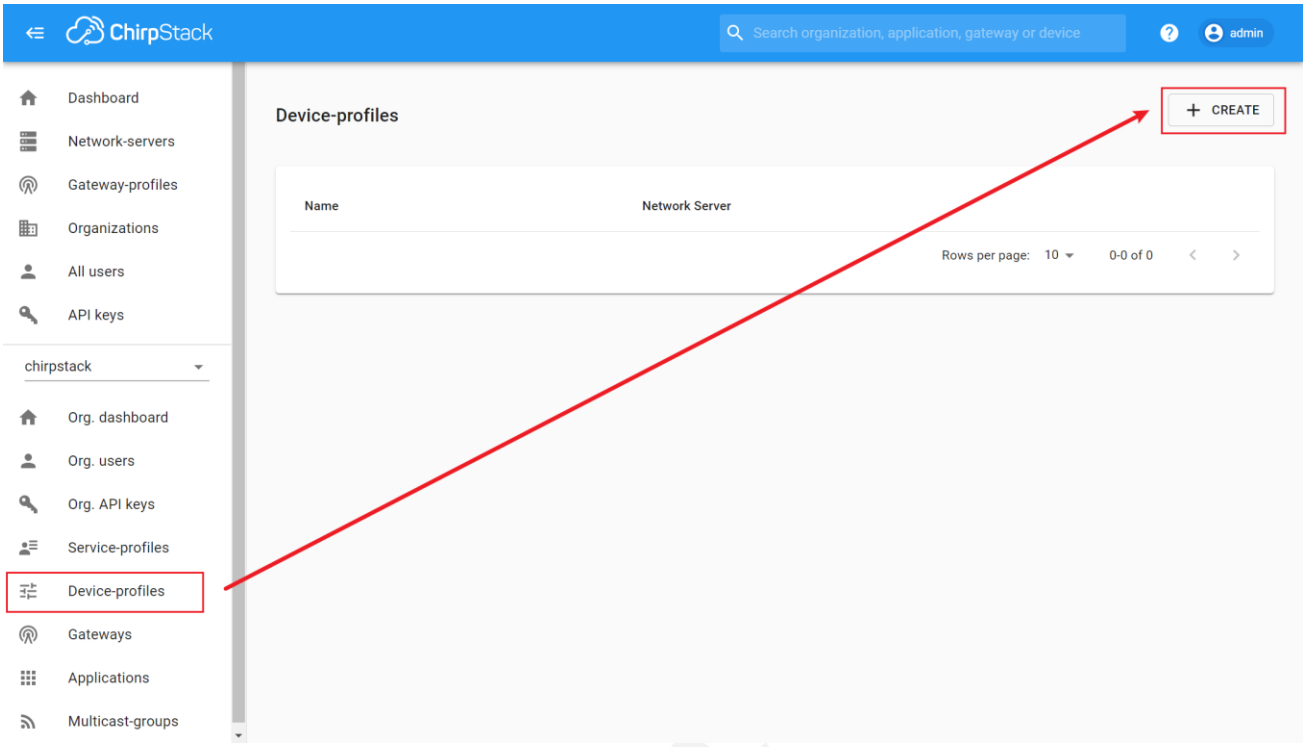
5.4.4 Create the Service-profiles

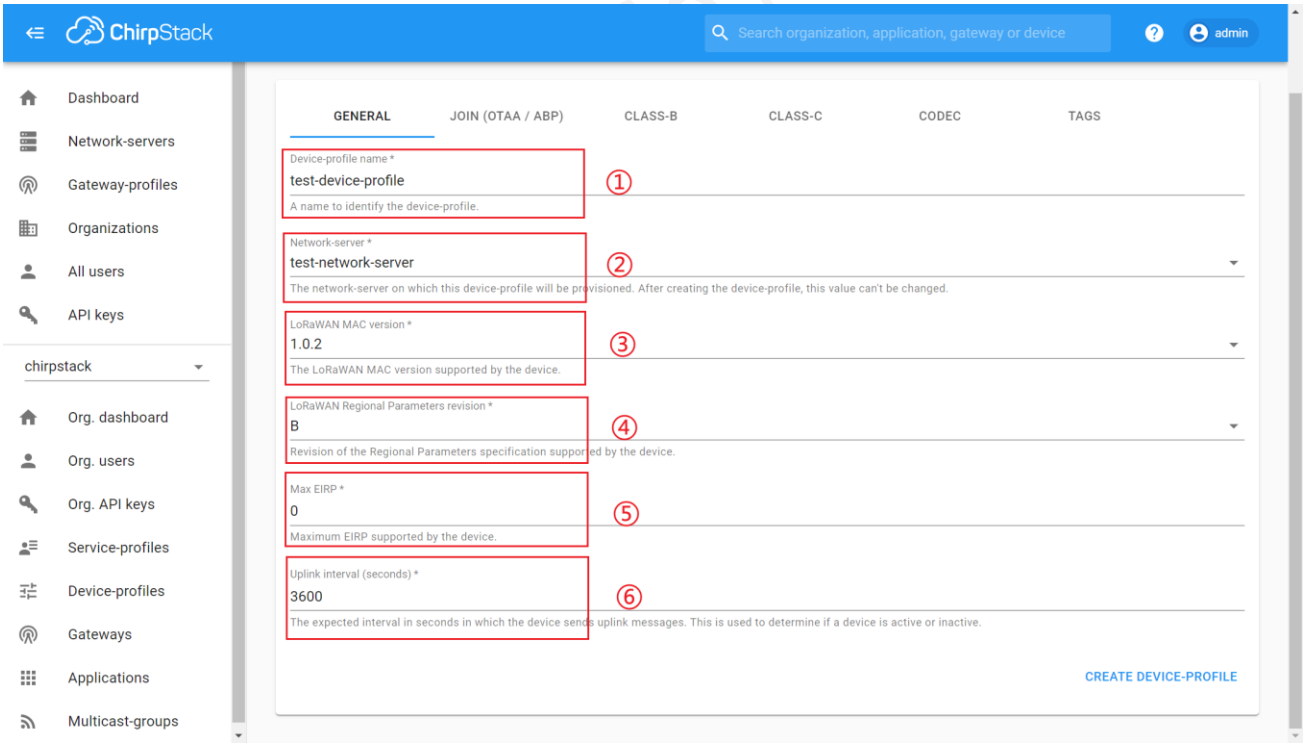




① Service-profile name: custom name.
② Network-server: select the Network-server you created earlier.
③ Add gateway meta-data: select it.
④ Default values are usually used.

5.4.5 Create the Device-profiles





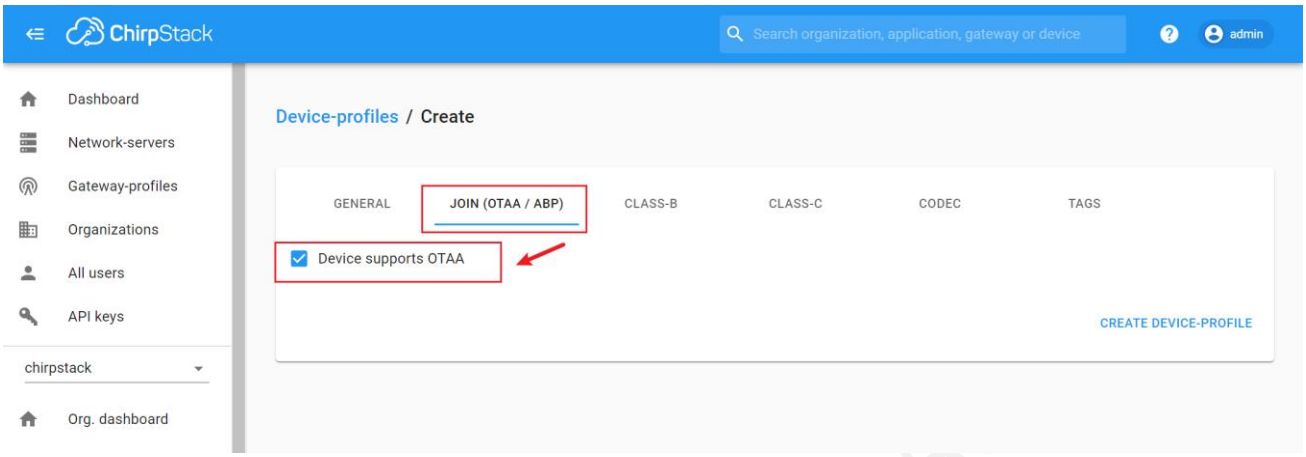
① Device-profile name: custom name.
② Network-server: select the Network-server you created earlier.
③ LoRaWAN MAC version: 1.0.2 (only for SenseCAP Node)
④ LoRaWAN Regional Parameters revision: B (only for SenseCAP Node)

⑤ Max EIRP: 0

⑥ Uplink interval (seconds) : 3600

Be consistent with the node's upload interval.

Click the “JOIN(OTAA/ABP)”, and select “Device supports OTAA”.



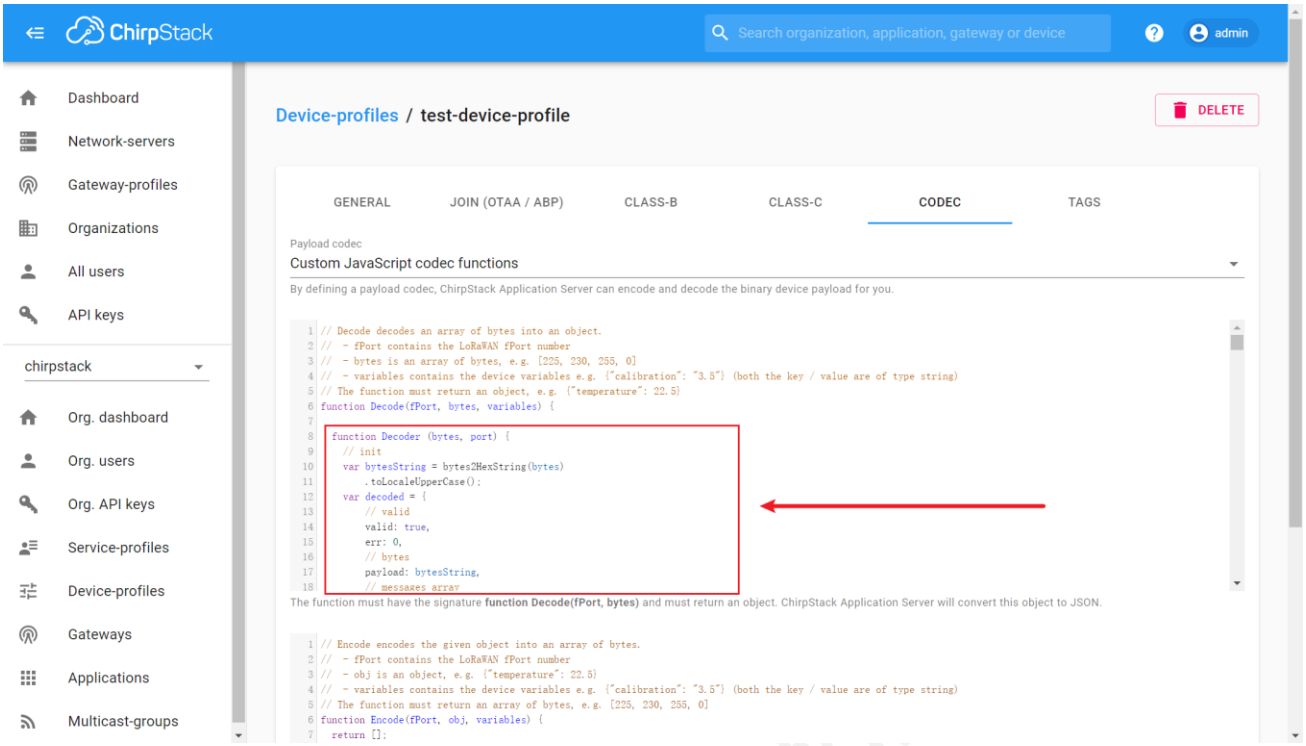
To get a SenseCAP Sensor Node on quick decoding, we provide a piece of code.

Click the “CODEC”, and select “Custom JavaScript codec functions”.

Then view https://github.com/Seeed-Solution/TTN-Payload-Decoder/blob/master/decoder.js , please copy the code to “function decode” FUNC.

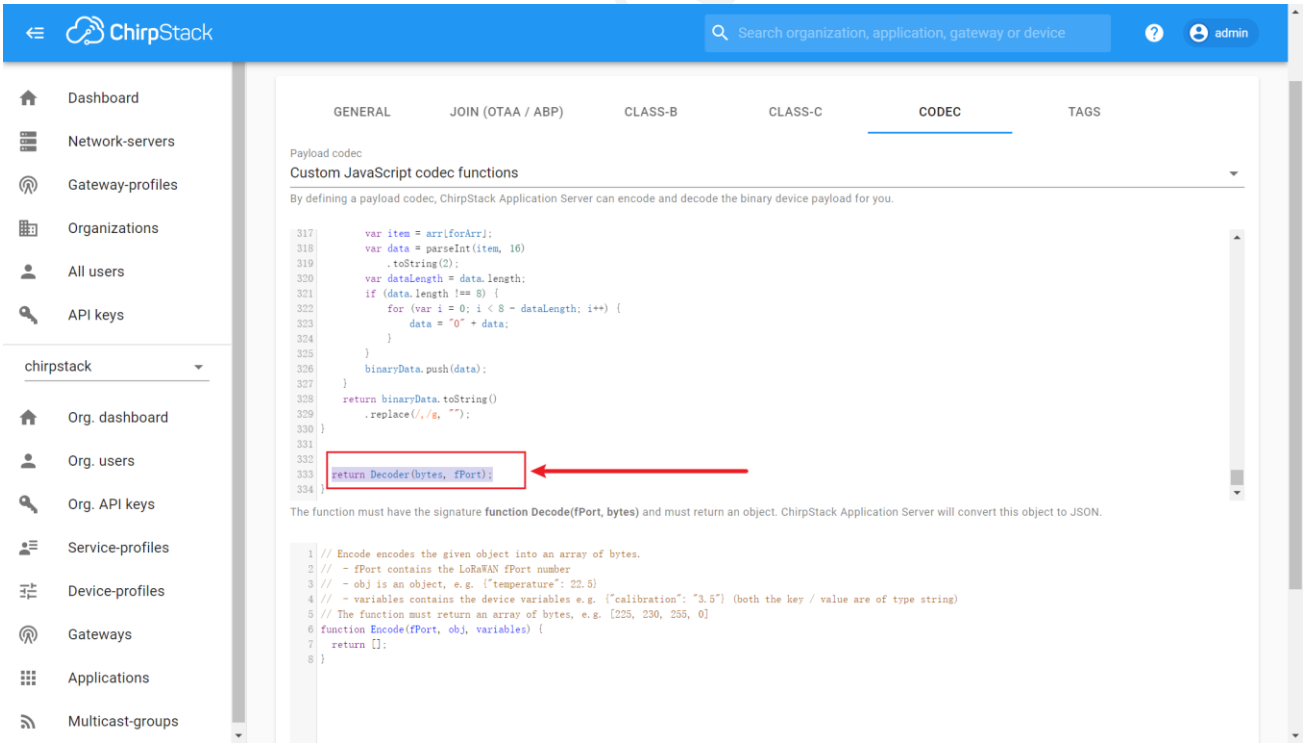
```
function Decoder (bytes, port) {
  // init
  var bytesString = bytes2HexString(bytes)
        .toLocaleUpperCase();
……..

return binaryData.toString()
          .replace(/,/g, "");
}
```



Add the return value at the end:

```
return Decoder(bytes, fPort);
```



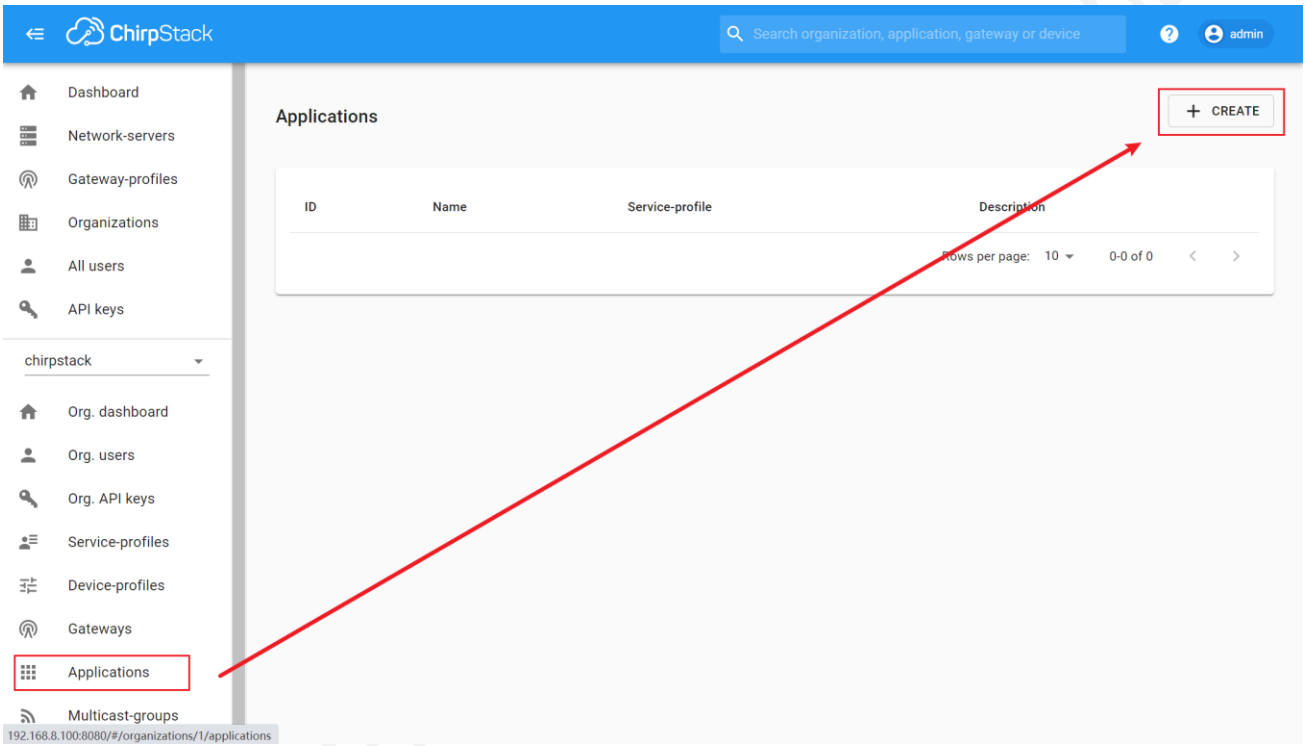
Finally, click “Create”.

5.5 Add Sensor Node to ChirpStack LoRa Server

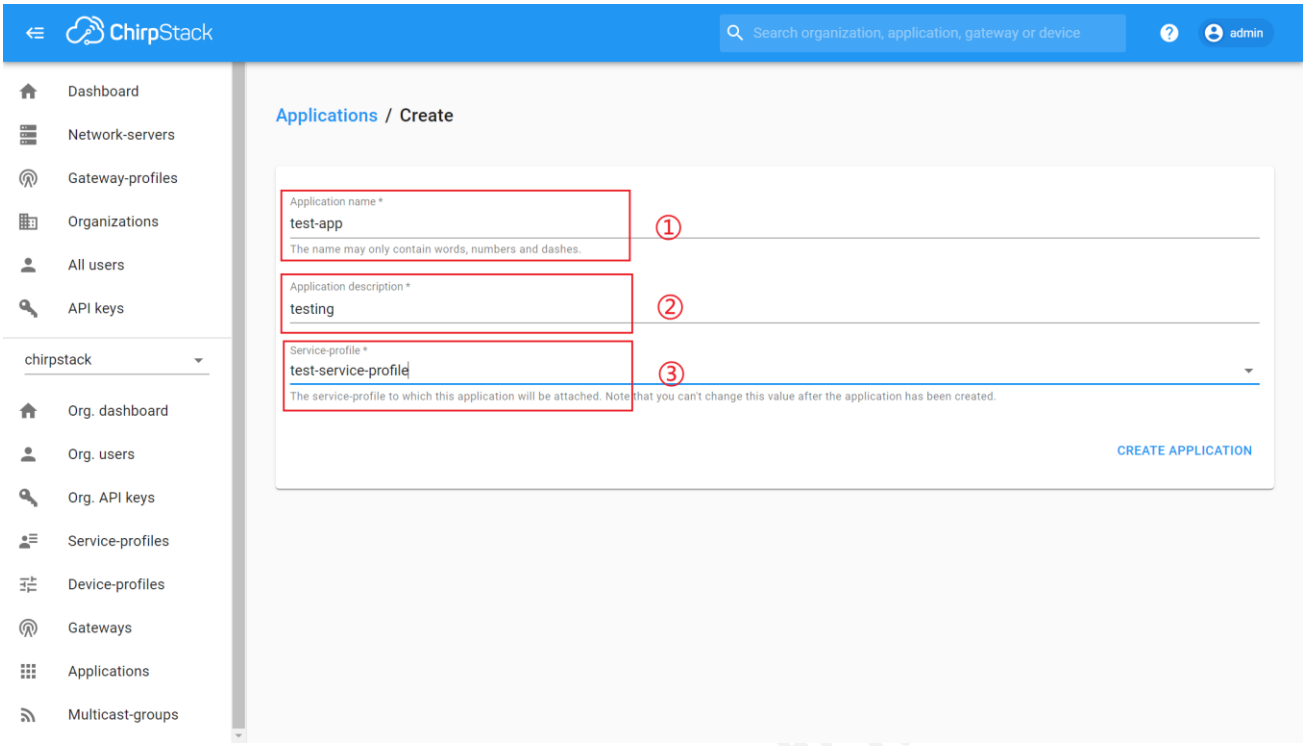
5.5.1 Get Node's EUI and Key

Refer to section 3.1.

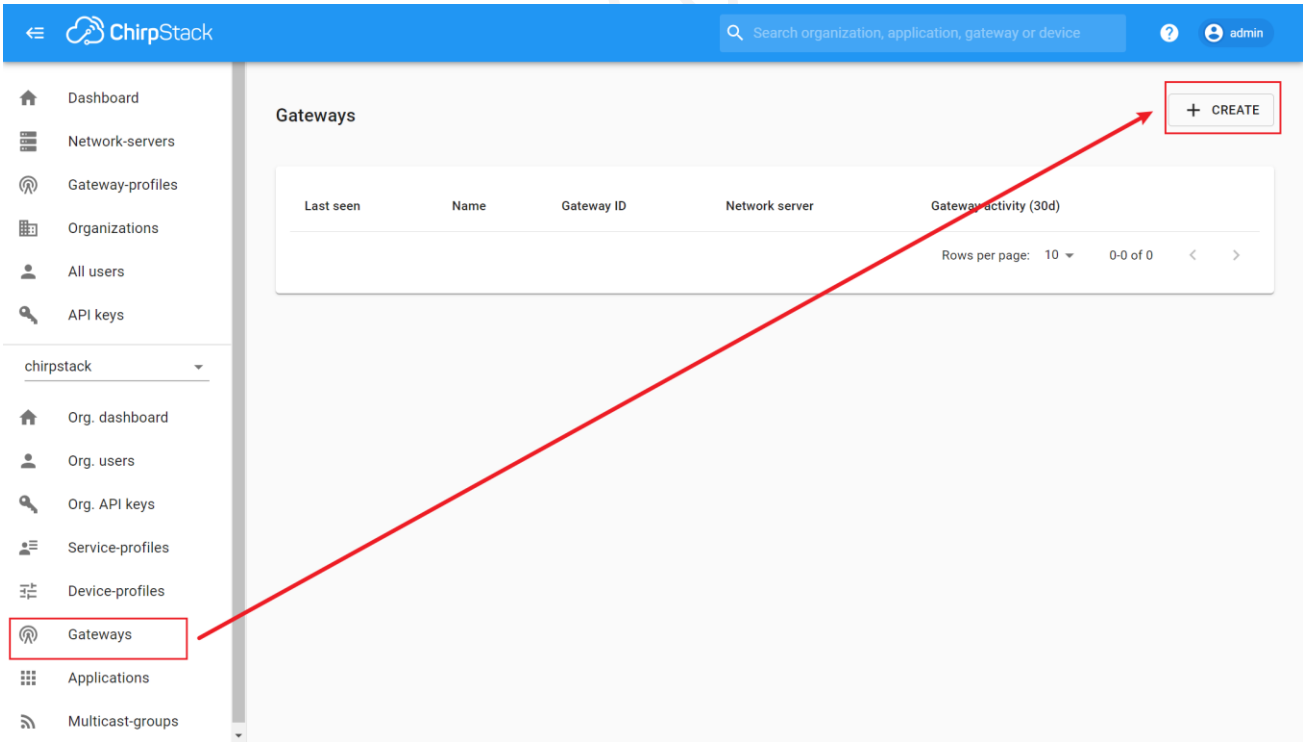
5.5.2 Create an Application

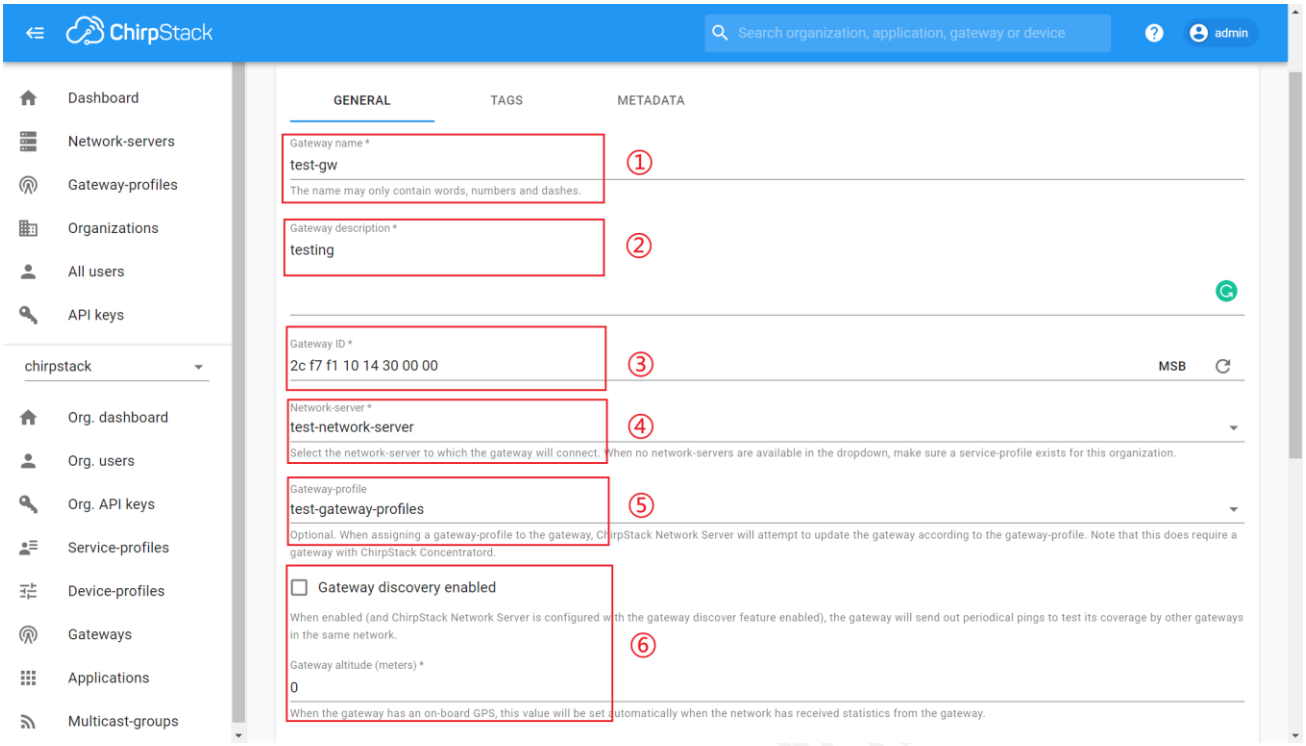


① Application name: custom name.
② Application description: custom description.
③ Service-profile: select the Service-profile you created earlier.



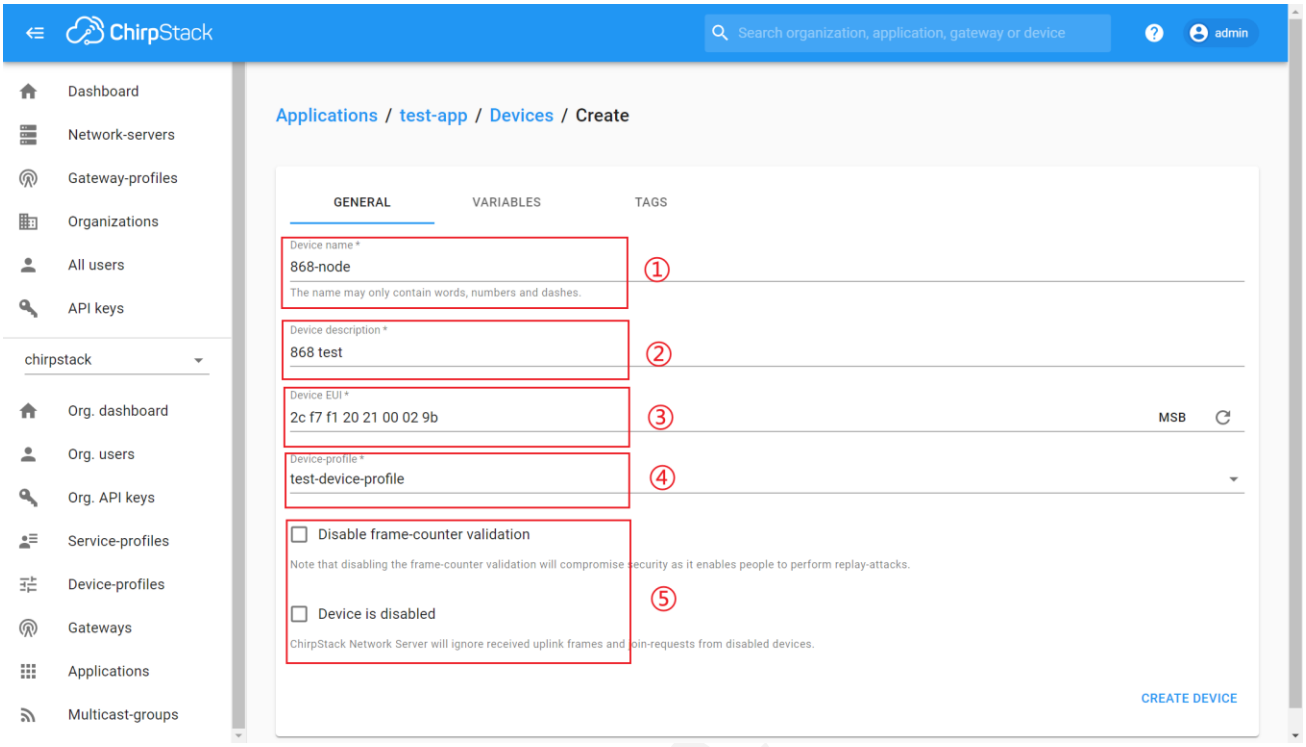
5.5.3 Create a Gateway





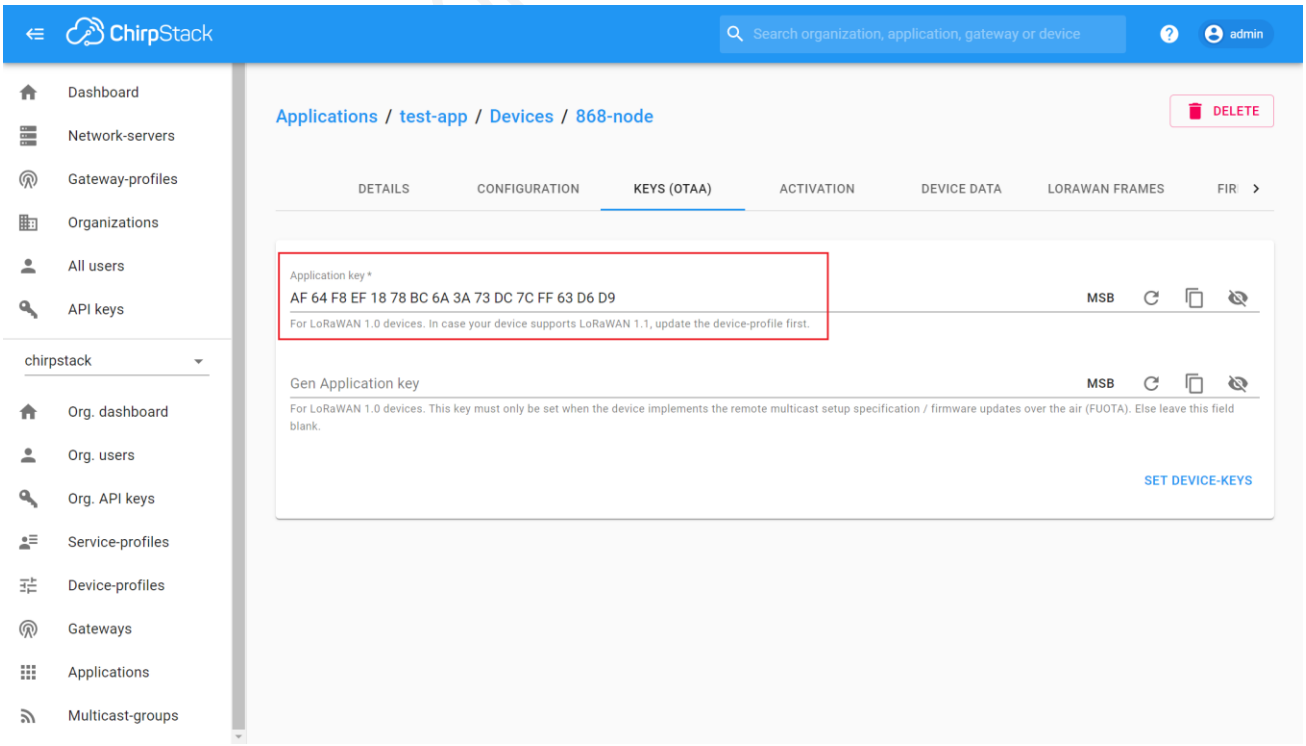
① Gateway name: custom name.
② Gateway description: custom description.
③ Gateway ID: the gateway EUI, see the gateway's label.
④ Network-server: select the Network-server you created earlier.
⑤ Gateway-profile: select the Gateway-profile you created earlier.
⑥ Default values are usually used.

5.5.4 Create a Device



① Device name: custom name.
② Device description: custom description.
③ Device EUI: the Node's EUI.
④ Device-profile: select the Device-profile you created earlier.
⑤ Don't check and ignore it.

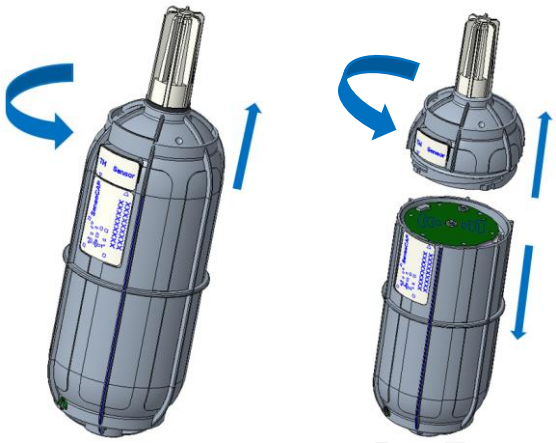
Click "Create" and enter the App KEY (Application Key, refer to section 3.1).



5.5.5 Power on

The power switch is hidden inside the device. Open the device and turn on the power before installing the sensors. Here is the step-by-step instruction:

4) Loosen the Sensor Probe by turning the cap counterclockwise. Use the white cap opener to make this process easier. The image below uses TH Sensor as an example and applies to all other SenseCAP sensors.



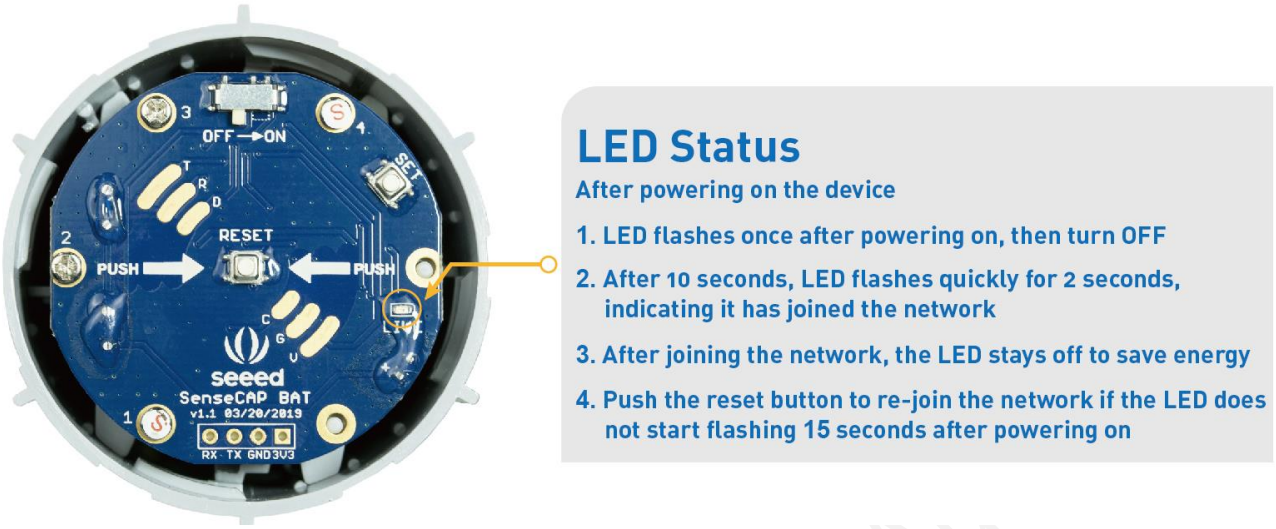
5) After opening the device, turn the switch to “ON”, and the LED on the lower right corner will flash, indicating that the power is on. Wait for about 10 seconds, then the LED will flash quickly for 2 seconds, indicating that the device is connected to the network.

6) After the device is connected to the network, connect the Sensor Probe back with the Sensor Node Controller by turning it clockwise. Please note that the labels on both parts should be aligned as shown in the image below, otherwise the two parts will not be attached to function properly and data will not be uploaded.

5.5.6 Sensor Node Working Status

You can refer to the LED indicator for the Sensor Node for its working status. Please see the status

explanations in the image below:

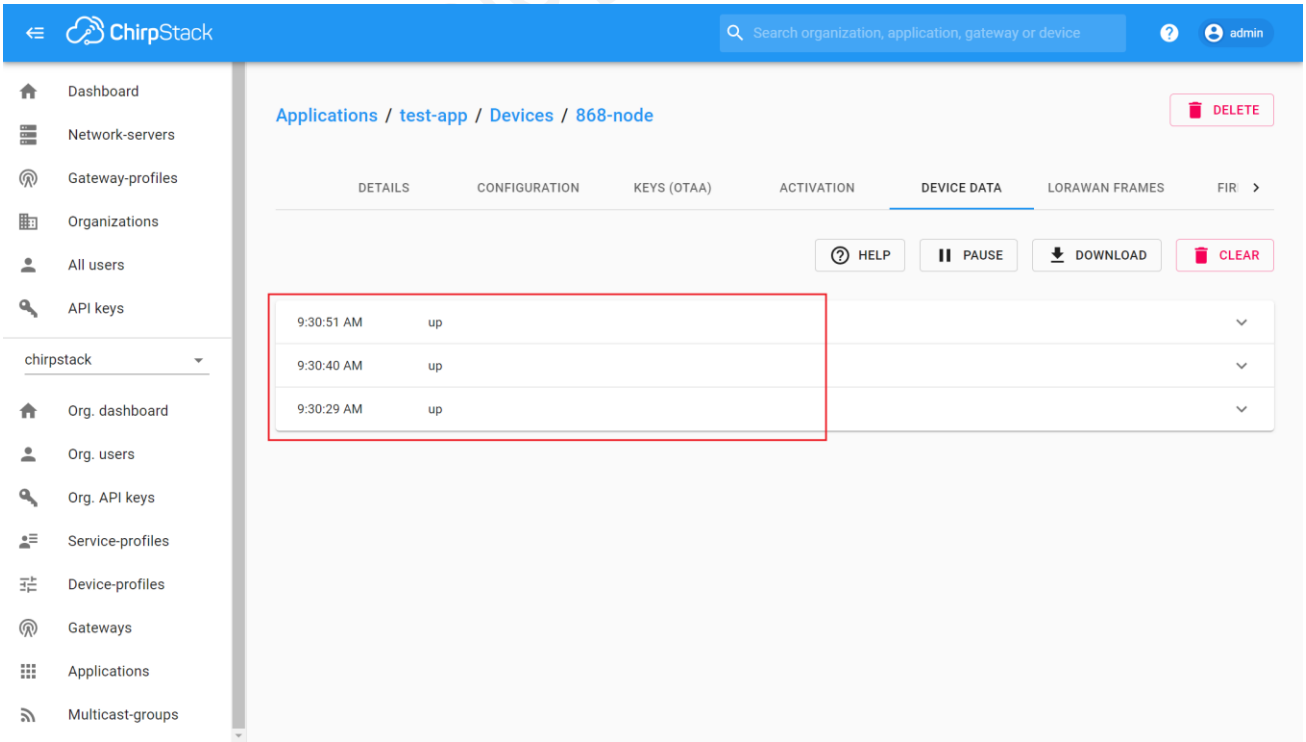


5.5.7 Checking Data Upload

On the “DEVICE DATA” page in the web, you can view the data that the gateway has received from the Sensor Node.

To get measurement ID information, please visit :

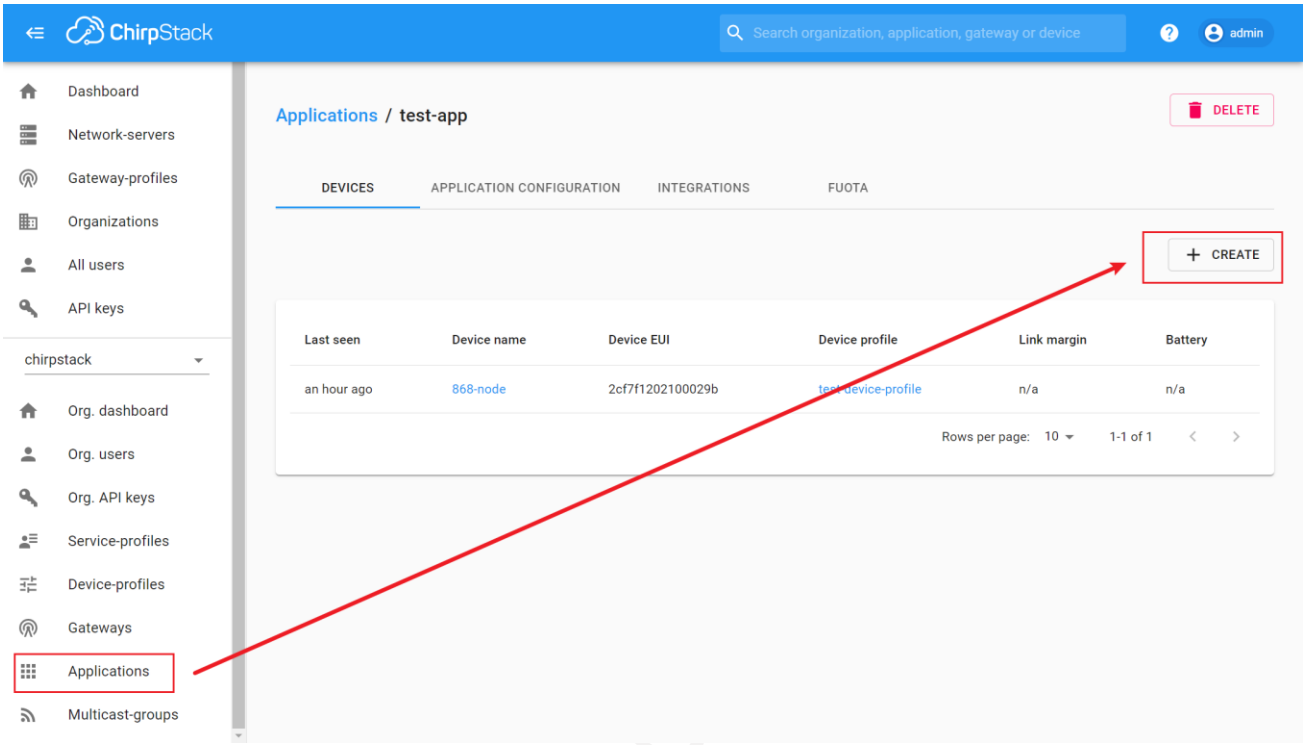
https://sensecap-docs.seeed.cc/sensor_types_list.html



5.6 Add a 3rd Part Node Device

(1) Refer to the previous section to configure the gateway.

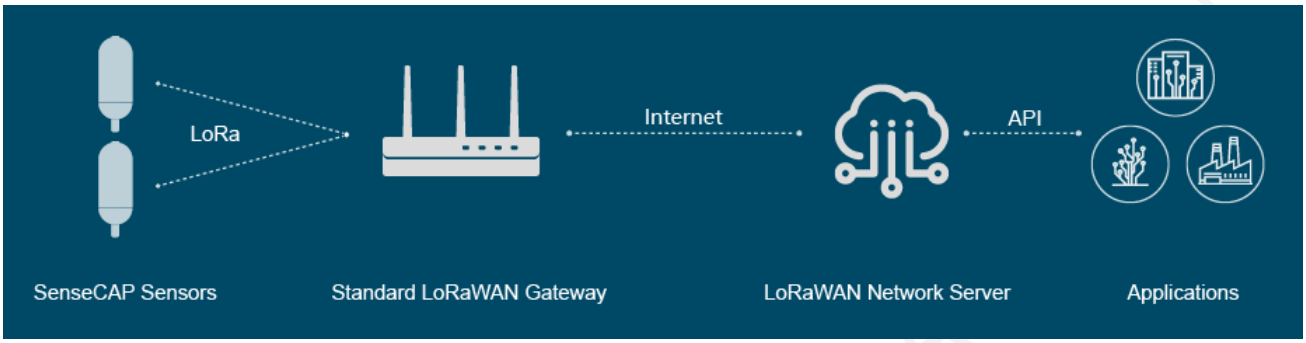
(2) Add a new device to Application.



(3) Refer to data parsing and tutorials for third-party devices.

6 The Node Connects to a Standard Gateway

SenseCAP Sensor Nodes are designed on The Things Network LoRaWAN servers, the firmware supports standard LoRaWAN 1.0.2 protocol, making it possible to connect to other 3rd-party LoRaWAN gateways and servers.



6.1 Node Frequency Plans

Frequency Plans	
EU868	Uplink: 868.1 - SF7BW125 to SF12BW125 868.3 - SF7BW125 to SF12BW125 and SF7BW250 868.5 - SF7BW125 to SF12BW125 867.1 - SF7BW125 to SF12BW125 867.3 - SF7BW125 to SF12BW125 867.5 - SF7BW125 to SF12BW125 867.7 - SF7BW125 to SF12BW125 867.9 - SF7BW125 to SF12BW125 868.8 – FSK Downlink: Uplink channels 1-9 (RX1) 869.525 - SF9BW125 (RX2 downlink only)
US915	Uplink: 903.9 - SF7BW125 to SF10BW125 904.1 - SF7BW125 to SF10BW125 904.3 - SF7BW125 to SF10BW125 904.5 - SF7BW125 to SF10BW125 904.7 - SF7BW125 to SF10BW125 904.9 - SF7BW125 to SF10BW125 905.1 - SF7BW125 to SF10BW125

	905.3 - SF7BW125 to SF10BW125 904.6 - SF8BW500 Downlink: 923.3 - SF7BW500 to SF12BW500 923.9 - SF7BW500 to SF12BW500 924.5 - SF7BW500 to SF12BW500 925.1 - SF7BW500 to SF12BW500 925.7 - SF7BW500 to SF12BW500 926.3 - SF7BW500 to SF12BW500 926.9 - SF7BW500 to SF12BW500 927.5 - SF7BW500 to SF12BW500

6.2 A Standard LoRaWAN Gateway Configuration Example

Typically, the LoRaWAN gateway needs to set the server address and uplink and downlink channel parameters for the end device. Refer to the gateway user manual to configure the server. Here, a common LoRaWAN gateway (US915) is taken as an example to explain how to configure the communication parameters of the Sensor Node.

The detailed configuration parameters for the Sensor Node are described here:
https://github.com/Jenkinlu001/SenseCAP-LoRaWAN/tree/master/LoRaWAN_Node_Parameters

6.2.1 Radio Settings

Find radio settings or frequency settings in the background of the gateway.

radio 0 enable√

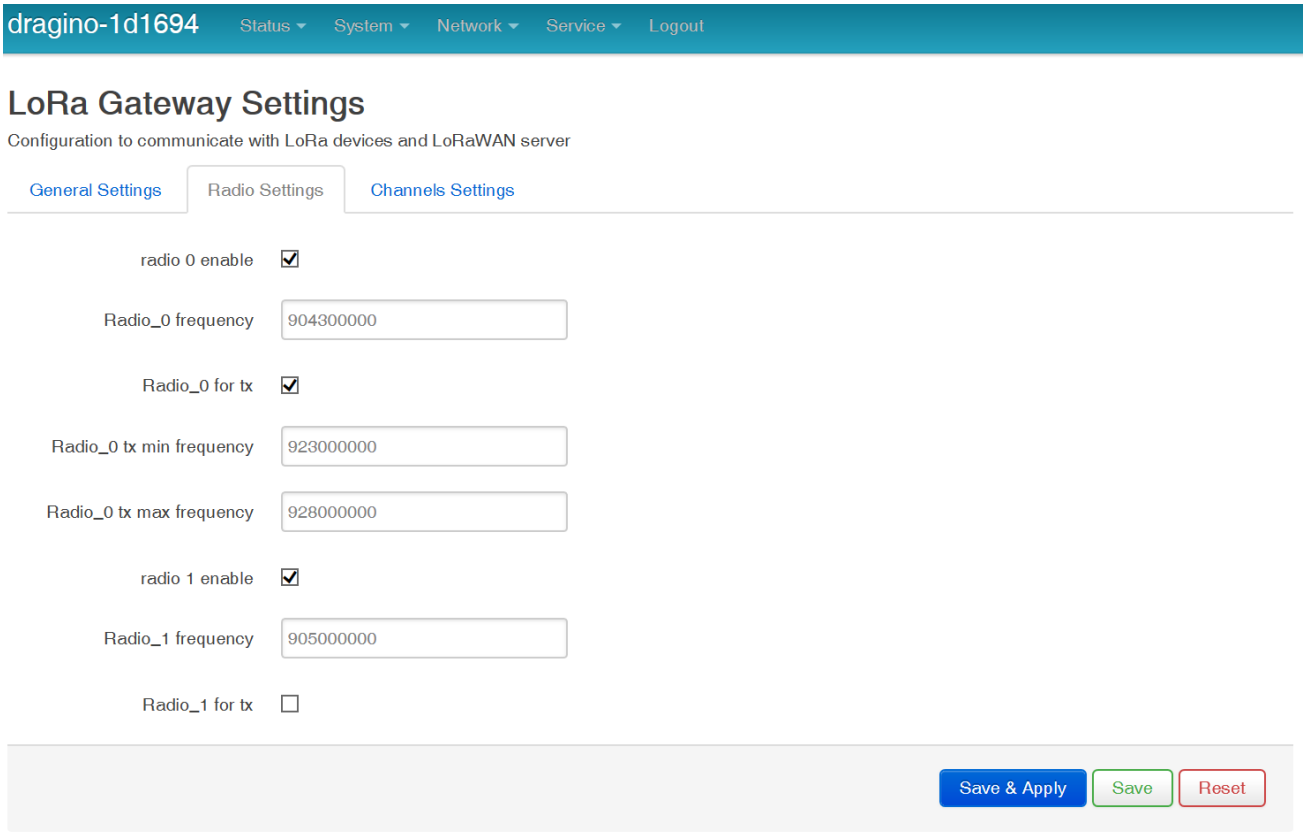
Radio_0 frequency: 904300000

Radio_0 for tx√

Radio_0 tx min frequency: 923000000
Radio_0 tx max frequency: 928000000

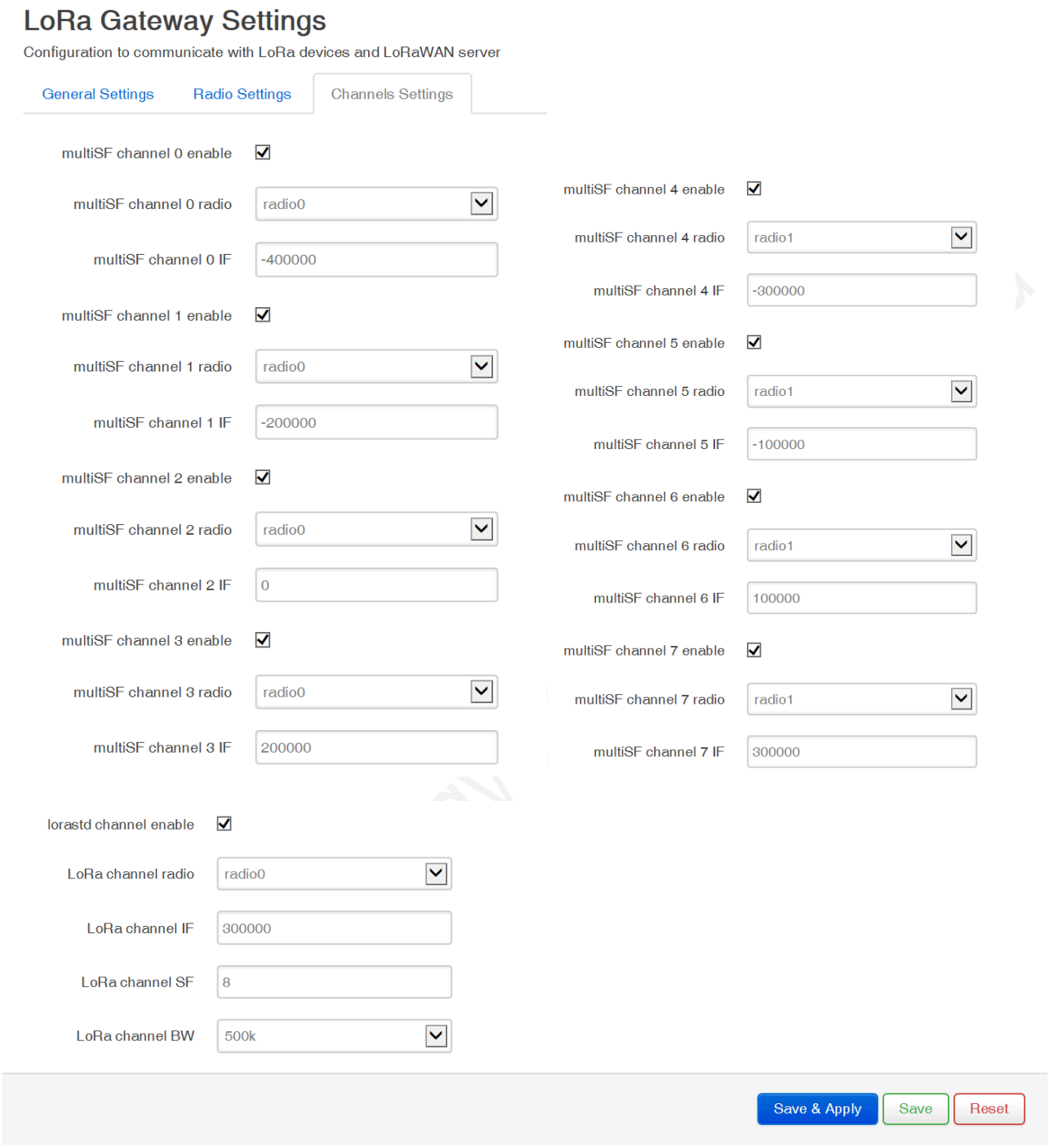
radio 1 enable√

Radio_1 frequency: 905000000



6.2.2 Channel Settings

Please refer to the items in the following image for channel settings.



6.2.3 Power on

Refer to section 4.5.5

6.2.4 Sensor Node Working Status

Refer to section 4.5.6

6.2.5 Checking Data Upload

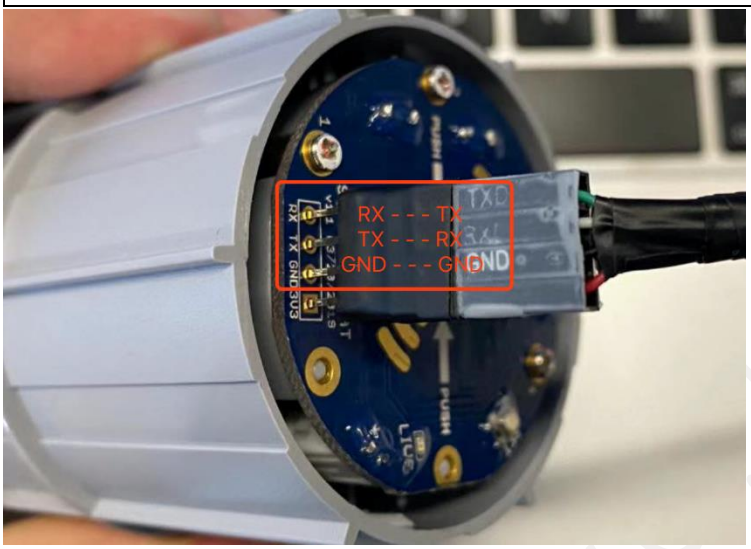
On the log page in the background of the gateway, you can view the data that the gateway has received from the Sensor Node.

6.3 Modify Node's EUI, KEY, and Duty

Connect serial ports (as shown in the image below), turn on the power, launch the serial port monitoring tool on your computer, set the Baud Rate as 115200.

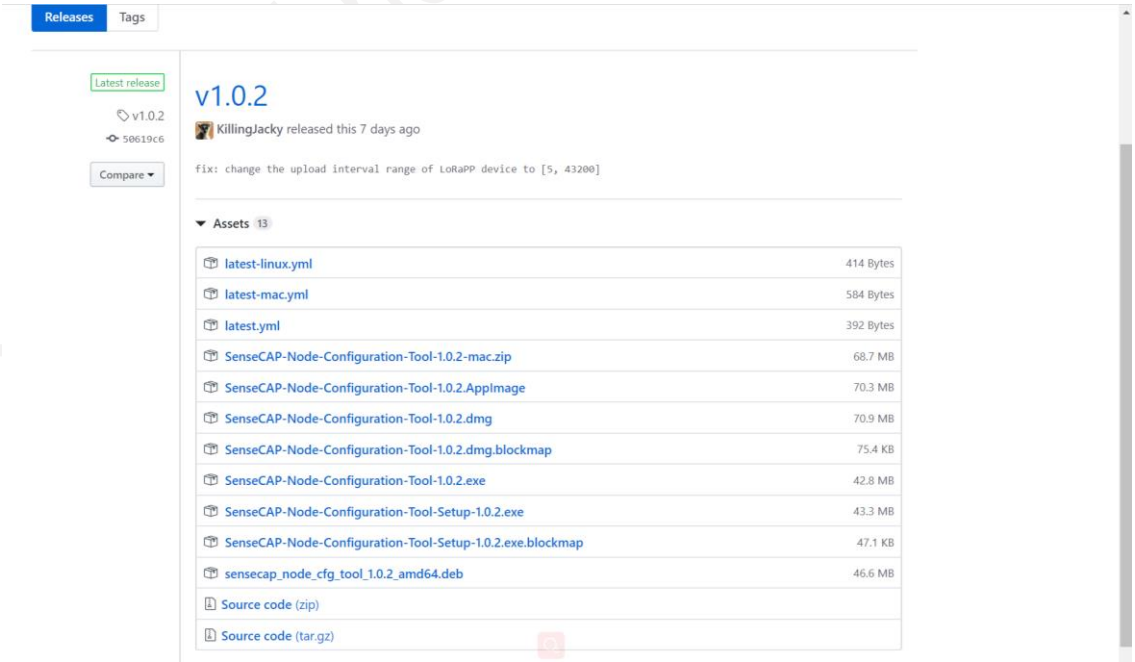
(1) Use the USB to TTL wire (Please leave power port, aka 3V3 unconnected):

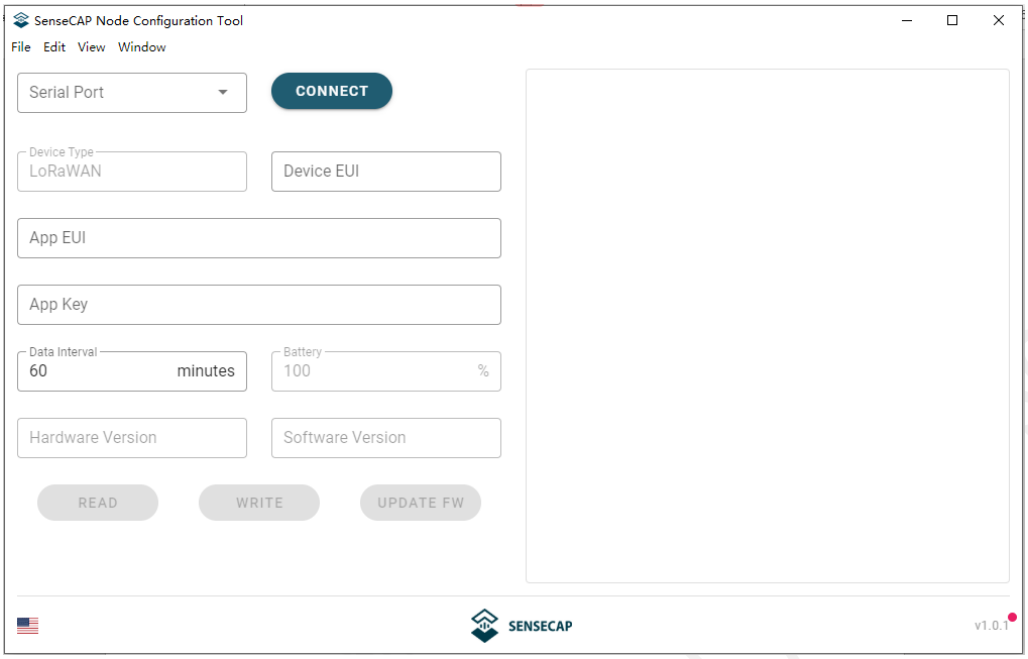
TX---RX RX---TX GND---GND



(2) Install the Serial Tool. Download via: https://github.com/Seeed-Solution/SenseCAP-Node-Configuration-Tool/releases/tag/v1.0.2

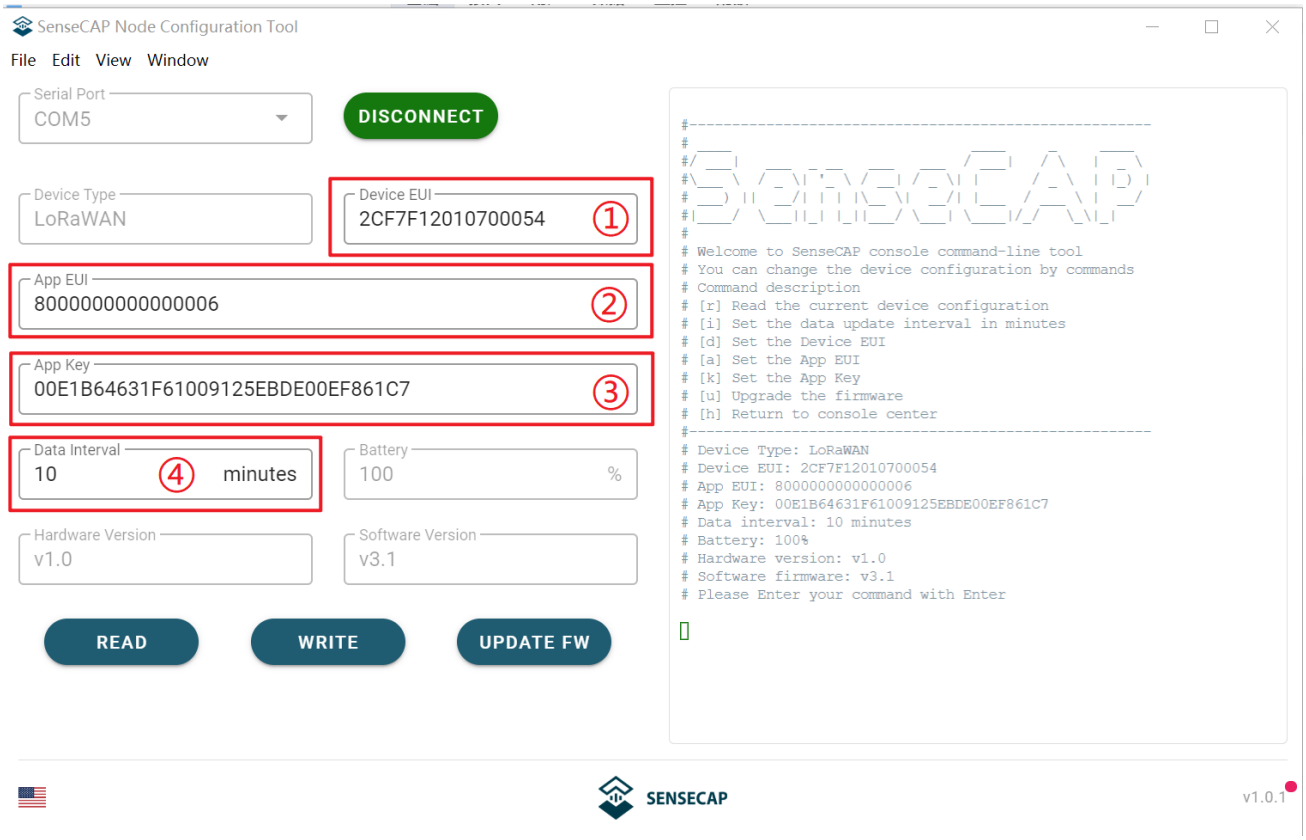
Windows: SenseCAP-Node-Configuration-Tool-1.x.x.exe
Mac: SenseCAP-Node-Configuration-Tool-1.0.2-mac.zip





(3) Select the COM Port that your tool uses, and click “CONNECT”.
Press “SET” button on the Sensor Controller, meanwhile flip the switch to “ON”, and you will see “SenseCAP”.

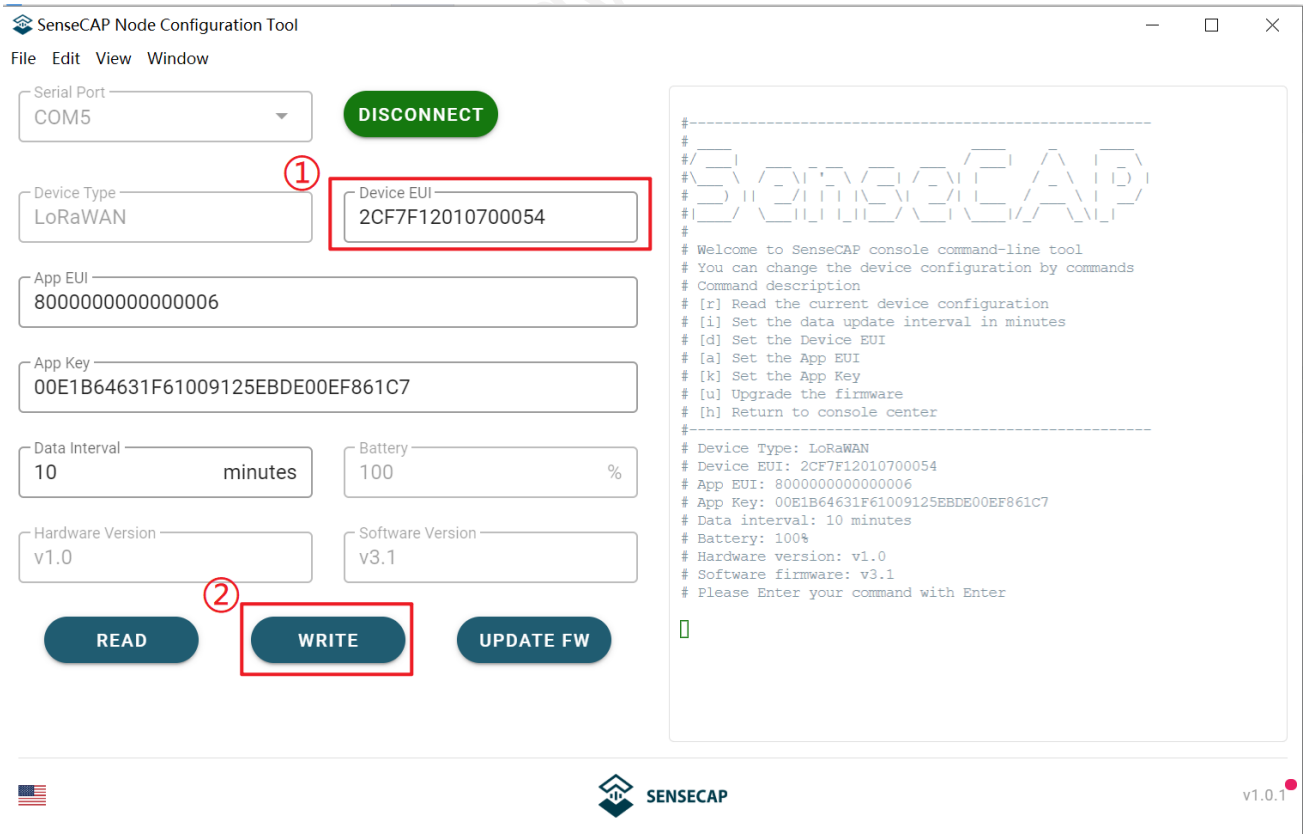
(4) ①Device EUI (16 bit) ②App EUI (16 bit) ③App Key (32 bit) ④Data Interval (Sensor collection cycle)



(5) For example: modify the Device EUI

① Write the new Device EUI.

② Click “WRITE”



(6) The Main Menu shows up, with respective commands. (Use other Serial Port Tool)

[r] Read the current device configuration
[i] Set the data update interval in minutes
[d] Set the Device EUI
[a] Set the App EUI
[k] Set the App Key
[u] Upgrade the firmware
[h] Return to console center

6.4 Modify the Data Interval Remotely

(1) Using the Network Server's portal or API to send downlink command, then the Node will respond to the ack.

Note: The downlink command takes effect and responds the next time the node uploads data.

(2) Select Port 2, Downlink as follow:

0x00	0x89	0x00	prepareId_L	prepareId_H	duty_L	duty_H	crc-L	crc-H

0x00	Fixed field
0x89	Fixed field
0x00	Fixed field
prepareId_L	Command ID low byte, you can customize the values, it allow each command ID to be the same
prepareId_H	Command ID high byte, you can customize the values, it allow each command ID to be the same
duty_L	Data interval low byte, you can set the data interval, unit: minute
duty_H	Data interval high byte, you can set the data interval, unit: minute
crc-L	CRC low byte, it's calculated by the CRC-16/CCITT
crc-H	CRC low byte, it's calculated by the CRC-16/CCITT

(3) When you send the downlink command, the Node responds to the ack command.

0x00	0x1F	0x00	prepareId_L	prepareId_H	result	0x00	crc-L	crc-H

0x00	Fixed field
0x1F	Fixed field
0x00	Fixed field
prepareId_L	Command ID low byte, it is the same as the downlink command
prepareId_H	Command ID high byte, it is the same as the downlink command
result	If the downlink command is in force, it responds 0x01, else it responds 0x00
0x00	Fixed field
crc-L	CRC low byte, it's calculated by the CRC-16/CCITT
crc-H	CRC low byte, it's calculated by the CRC-16/CCITT

For example: Set the Node's data interval is 10 minutes.

Send the downlink command (HEX):

00 89 00 11 22 0A 00 38 B4

0x00	0x89	0x00	prepareId_L	prepareId_H	duty_L	duty_H	crc-L	crc-H
00	89	00	11	22	0A	00	38	B4

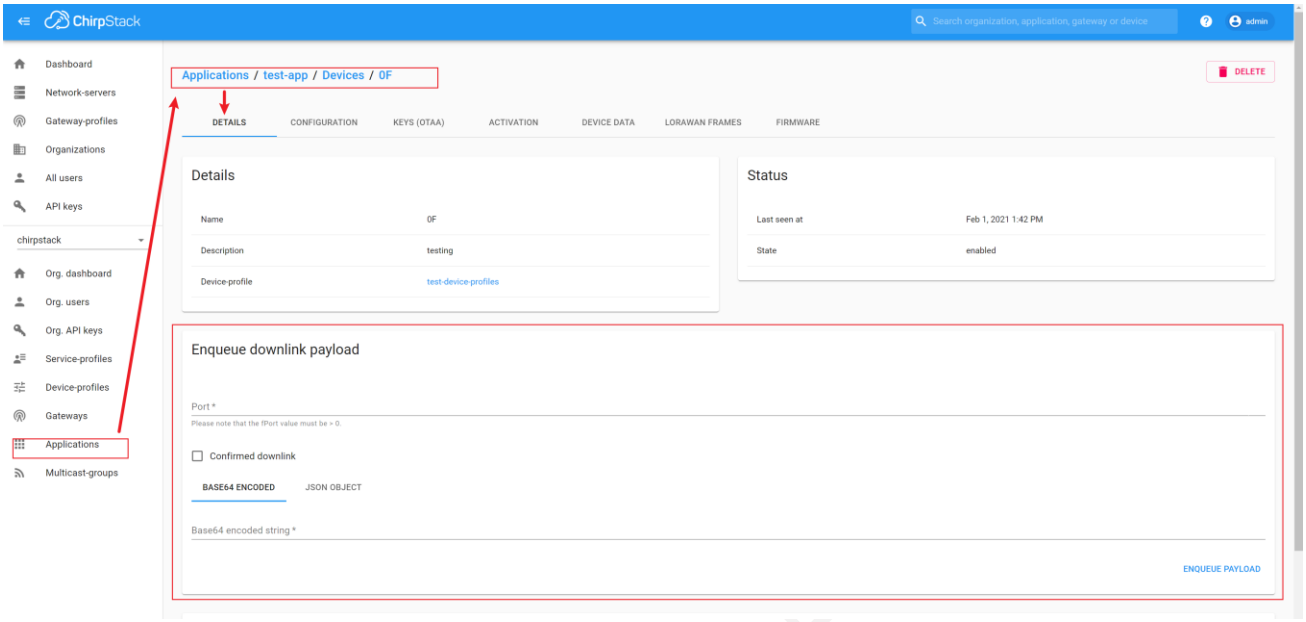
ACK Response:

00 1F 00 11 22 01 00 78 0F

0x00	0x1F	0x00	prepareId_L	prepareId_H	result	0x00	crc-L	crc-H
00	1F	00	11	22	01	00	78	0F

6.4.1 Modify the Data Interval via the Chirpstack

(1) Click to “Application→Devices→Node→DETAILS”



(2) Enqueue downlink payload:

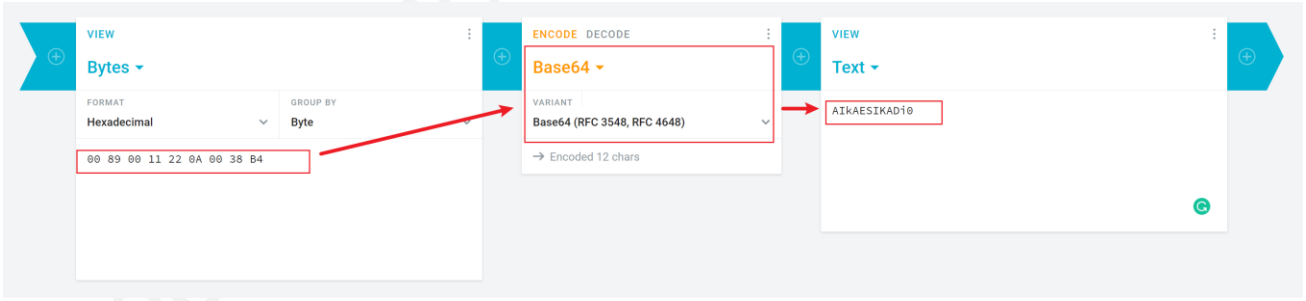
a) Port: 2

b) Select “Confirmed downlink”.

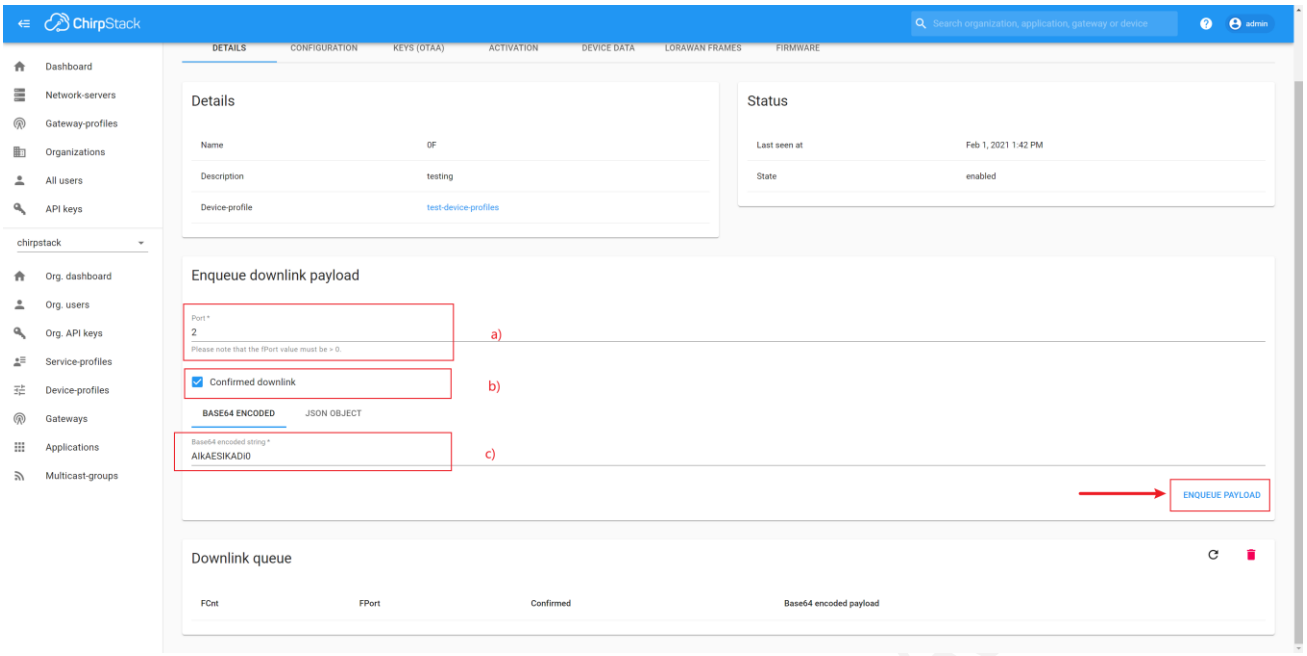
c) Input the Base64 command,

Set the Node’s data interval is 10 minutes, and send the downlink command (HEX): **00 89 00 11 22 0A 00 38 B4**

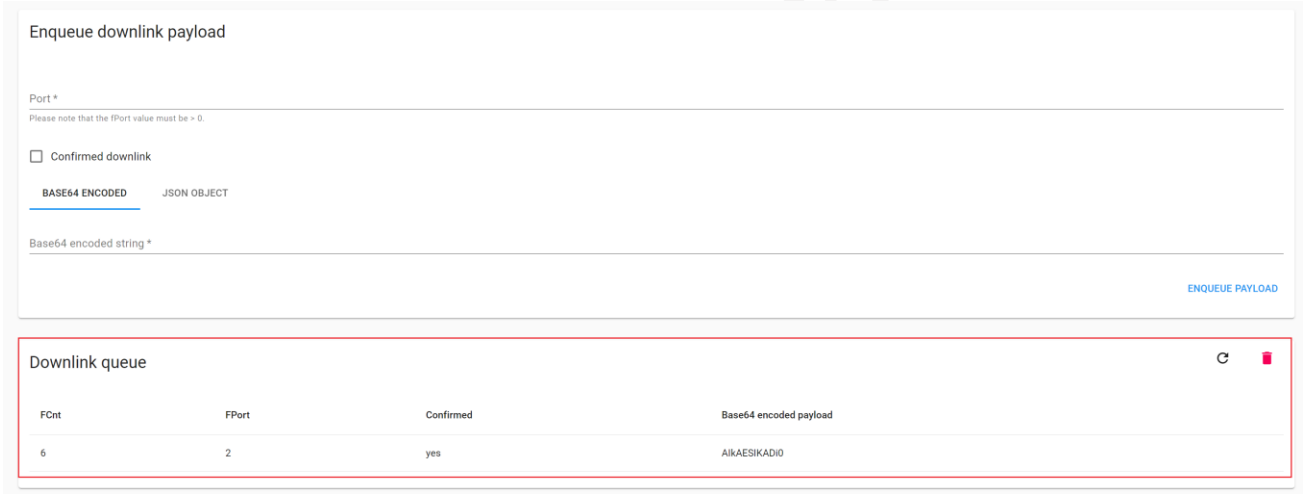
Then, use a hex to base64 tool (https://cryptii.com/pipes/hex-to-base64).



So, the base64 command is **AIkAESIKADi0**

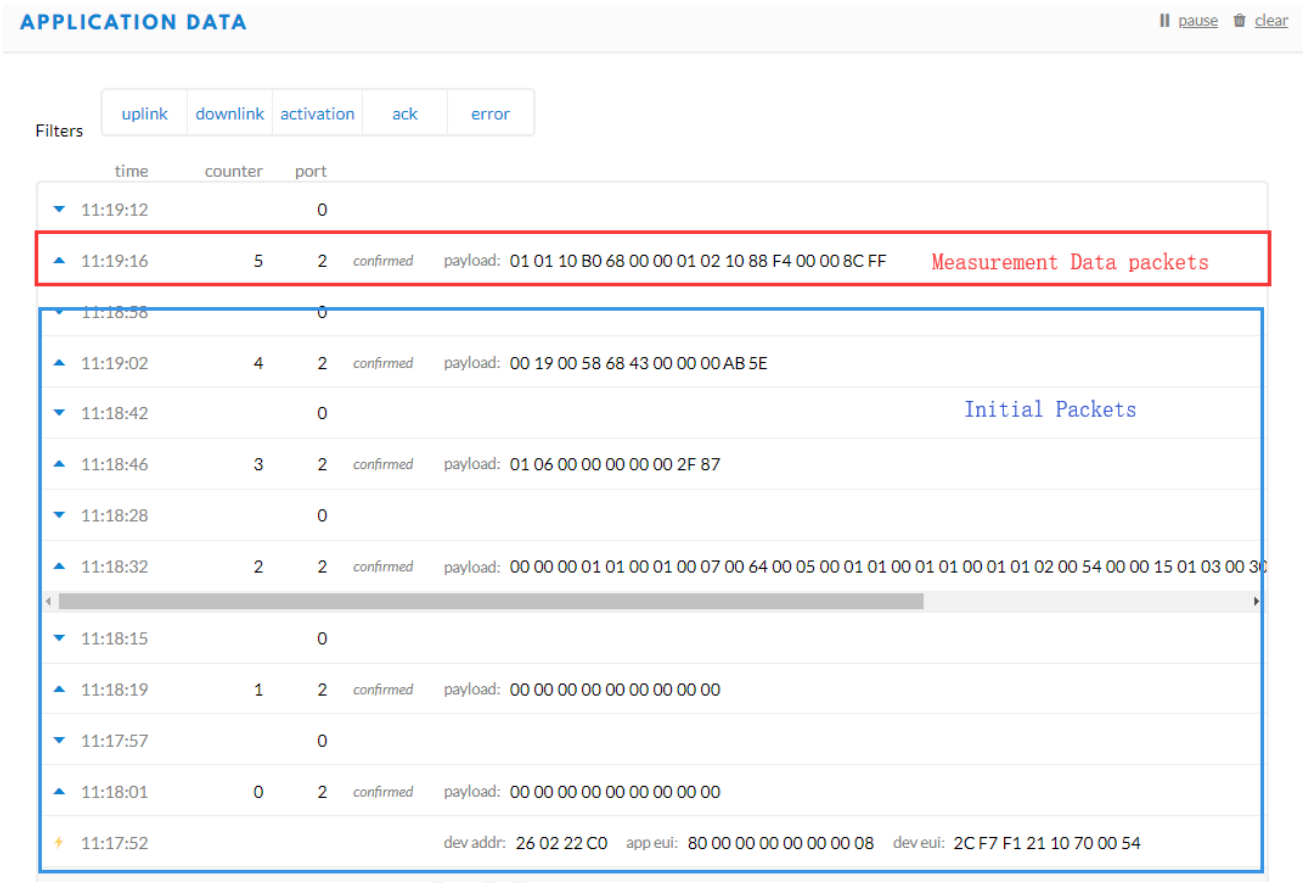


d) Click the “ENQUEUE PAYLOAD”, the ”downlink queue” will display command.
When the command disappears after you refresh, the command has been sent.



7 Decoding

In the gateway or server background, similar packets can be viewed.(If the data is encrypted, it usually needs to be decrypted using base64)



Notice:

With successful access to the network, please connect the Sensor Probe back to the Sensor Node Controller by turning it clockwise. Please note the labels on both sides should be aligned as the image below, or it will not be put back in the right way. When the Sensor Probe is connected to the Sensor Node Controller correctly, the device can upload data.

7.1 Packet Parsing

Packet Initialization

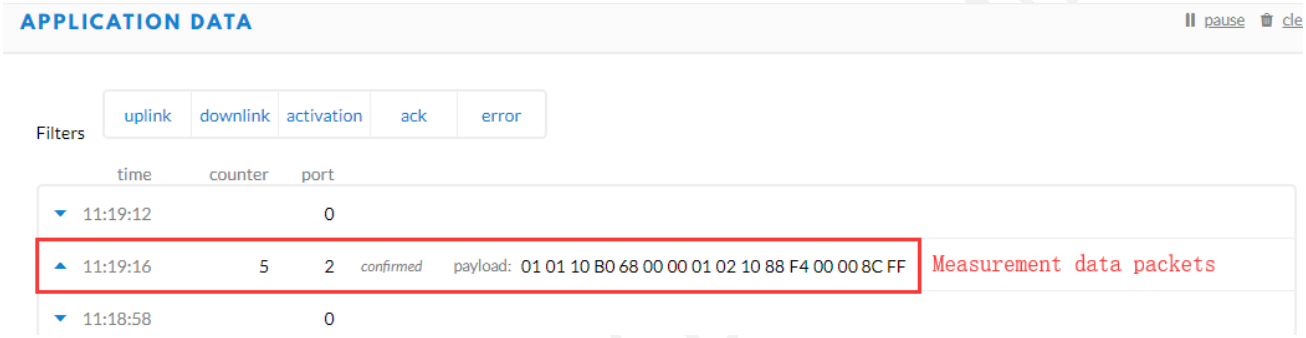
After being powered on or reboot, SenseCAP Sensor Nodes will be connected to the network using OTAA activation method. Each Sensor Node will send data packets to the server, including the following data:

Initial packets (no need to learn about these initial packets)

- One packet with device info including hardware version, software version, battery level, sensor hardware & software version, sensor EUI, power, and sensor power time counter at each channel.

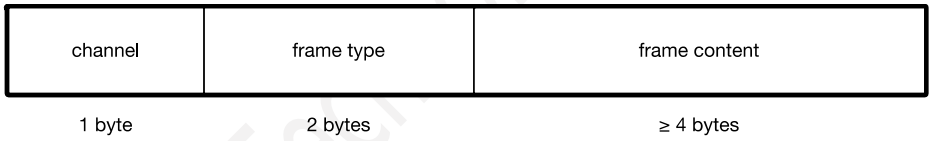
Measurement data packets

The only thing we should pay attention to is the sensor measurement data packets



Packet Structure

The structure of the frame is shown in the image below.



1 byte for channel, default as 1, means the sensor has been well connected.

2 bytes for frame type, in this case, it will be 0110 and 0210, means temperature value and humidity value

4 bytes for content, is the sensor value with CRC

The frame content is sent in little-endian byte order

7.1.1 Example 1 - Air Temperature & Humidity Sensor:

Air Temperature & Humidity Sensor measurement packet: 010110B0680000010210 88F400008CFF

Divide the data into 3 sections

1	Air Temperature	010110B0680000	01 is the channel number. 0110 is 0x1001 *(little-endian byte order)* , which is the measurement ID for air temperature. B0680000 is actually 0x000068B0, whose equivalent decimal value is 26800. Divide it by 1000, and you' ll get the actual measurement value for air temperature as **26.8℃**.
2	Air Humidity	01021088F40000	0210 is 0x1002 *(little-endian byte order)* , which is the measurement ID for air humidity. 88F40000 is actually 0x0000F488, whose equivalent decimal value is 62600. Divide it by 1000, and you' ll get the actual measurement value for air humidity as 62.6%RH.
3	CRC	8CFF	The CRC verification part.

7.1.2 Example 2 - CO2 Sensor:

CO2 Sensor measurement packet: 010410E08D05009802

Divide the data into 3 sections

1	CO2	010410E08D0500	01 is the channel number. 0410 is 0x1004 *(little-endian byte order)* , which is the measurement ID for CO2.

			E08D0500 is actually 0x00058DE0, whose equivalent decimal value is 364000. Divide it by 1000, and you' ll get the actual measurement value for CO2 as **364ppm**.
3	CRC	9802	The CRC verification part.

7.1.3 Example 3 - Soil Moisture and Temperature Sensor:

Soil Moisture and Temperature Sensor measurement packet: 010610007D0000010710725100009A21

Divide the data into 3 sections

1	Soil Temperature	010610007D0000	01 is the channel number. 0710 is 0x1007 *(little-endian byte order)* , which is the measurement ID for soil temperature. 007D0000 is actually 0x00007D00, whose equivalent decimal value is 32000. Divide it by 1000, and you' ll get the actual measurement value for Soil Temperature as 32.0℃.
2	Soil Moisture	010710725100000	0710 is 0x1007 *(little-endian byte order)* , which is the measurement ID for soil moisture. 72510000 is actually 0x00005172, whose equivalent decimal value is 20850. Divide it by 1000, and you' ll get the actual measurement value for Soil

			Moisture as 20.85%.
3	CRC	9A21	The CRC verification part.

7.1.4 Example 4 – Light Intensity Sensor:

Light Intensity Sensor measurement packet: 010310A0320000C3B6

Divide the data into 3 sections

1	Light Intensity	010310A0320000	01 is the channel number. 0310 is 0x1003 *(little-endian byte order)* , which is the measurement ID for Light Intensity. A0320000 is actually 0x000032A0, whose equivalent decimal value is 12960. Divide it by 1000, and you' ll get the actual measurement value for Light Intensity as 12.96**Lux**.
3	CRC	C3B6	The CRC verification part.

7.1.5 Example 5 – Barometric Pressure Sensor:

Barometric Pressure Sensor measurement packet: 010510284A140652B7

Divide the data into 3 sections

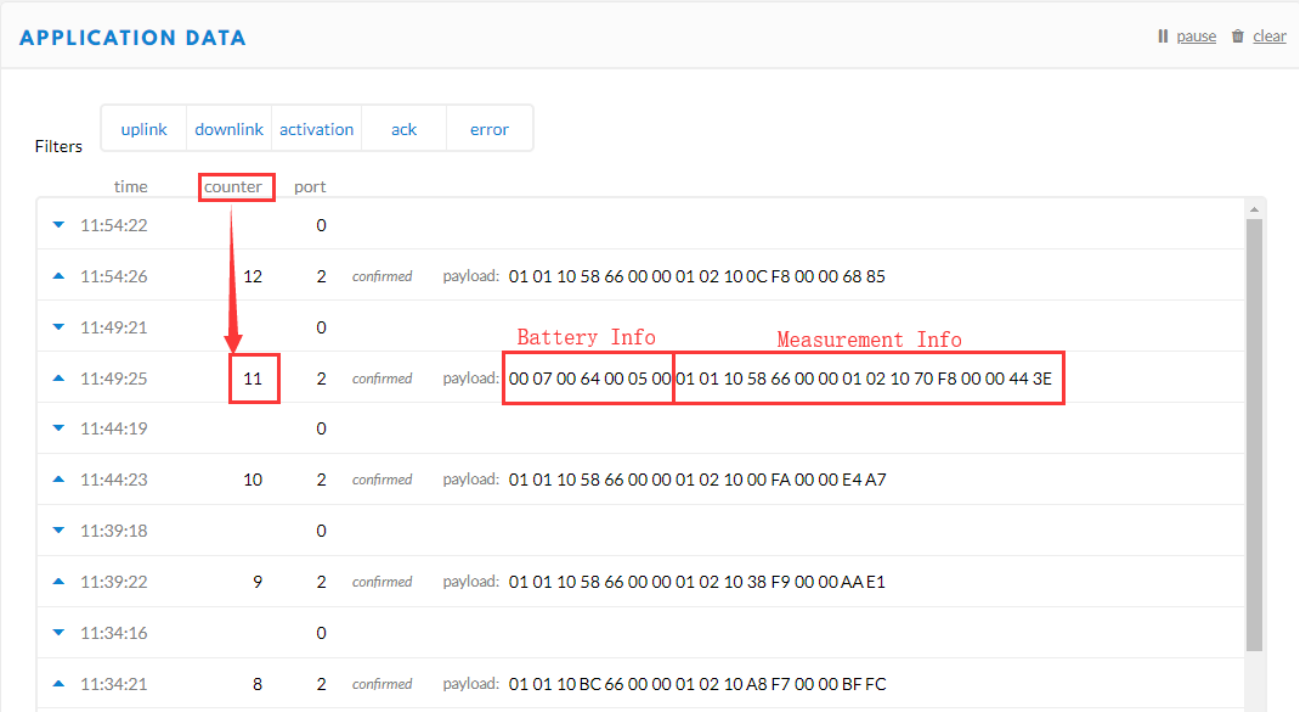
1	Barometric Pressure	010510284A1406	01 is the channel number. 0510 is 0x1003 *(little-endian byte order)* , which is the measurement ID for Barometric Pressure.

			284A1406 is actually 0x06144A28, whose equivalent decimal value is 101993000. Divide it by 1000, and you' ll get the actual measurement value for Barometric Pressure as 101993**Pa**.
3	CRC	52B7	The CRC verification part.

To get more measurement ID, please visit https://sensecap-docs.seeed.cc/sensor_types_list.html

7.2 Exception

Please note the counter number. After 10 packets, it will follows one special packet with battery info. You can either ignore this packet or get rid of the battery info in your code.



Original Info: 0007006400050001011058660000010210 70F80000443E

Battery Info: 00070064000500

Measurement Info: 0101105866000001021070F80000443E

Example:

Battery & TH Sensor measurement packet: 0007006400050001011058660000010210 70F80000443E

Divide the data into 3 sections

1	Battery	00070064000500	
2	Temperature	0101105866000 0	01 is the channel number. 0110 is 0x1001 *(little-endian byte order)* , which is the measurement ID for air temperature.

			58660000 is actually 0x00006658, whose equivalent decimal value is 26200. Divide it by 1000, and you' ll get the actual measurement value for air temperature as **26.2℃**.
2	Humidity	01021070F80000	0210 is 0x1002 *(little-endian byte order)* , which is the measurement ID for air humidity. 70F80000 is actually 0x0000F870, whose equivalent decimal value is 63600. Divide it by 1000, and you' ll get the actual measurement value for air humidity as 63.6%RH.
3	CRC	443E	The CRC verification part.

8 Device Installation

In this chapter, we will introduce the gateway and sensor nodes, their respective installation processes, as well as the dos and don’ts. Before installing, please check the part list to ensure nothing is missing.

8.1 Part List

8.1.1 Gateway Part List

The LoRa Gateway comes with a standard antenna. If you need ultra-long-distance communication, you will need to purchase a high-gain fiberglass antenna.

Item	Name	Quantity
1	LoRa Gateway	1
2	LoRa Antenna	1
3	4G Antenna	1
4	Allen Hex Key	1
5	Mounts	4
6	Power Adapter	1
7	Power Extension Cable (5M)	1
8	Ferrules / Aluminum piece	2 / 2
9	M5 Self-drilling Screw	8
10	Antenna Lightning Protector (*Optional)	1
11	LoRa Fiberglass Omni Antenna (*Optional)	1
12	LoRa Antenna Brackets (*Optional)	1

8.1.2 Sensor Node Part List

The accessories for different sensors may vary. The common parts are as follows:

Item	Name	Quantity
1	Sensor	1
2	Bracket	1
3	M4 Self-drilling Screw	4
4	M3 Self-drilling Screw	2

8.1.3 Other Accessories & Tool List

For installing in different scenarios, you might need to purchase extra accessories or tools.

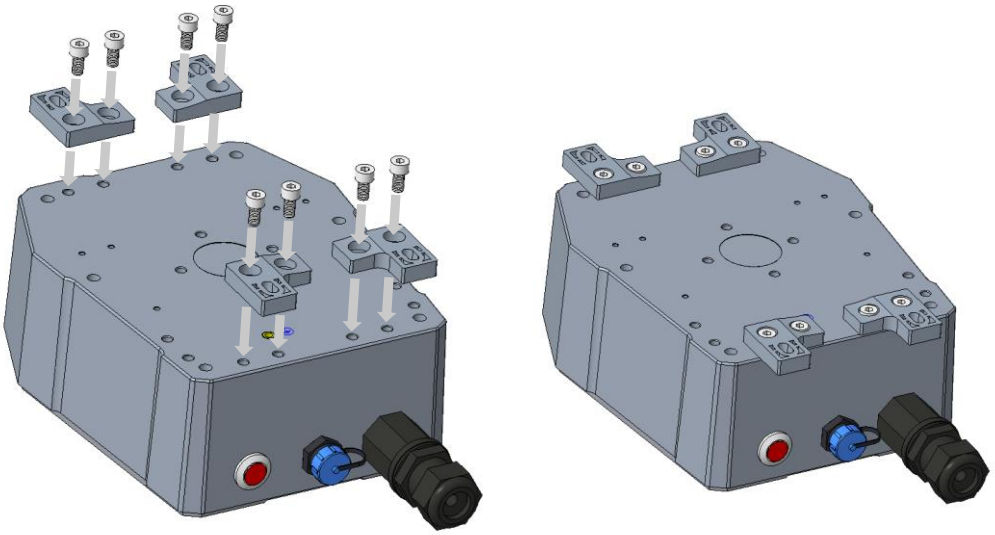
Item	Name	Quantity
1	GND Copper Wire (2.5mm^2)	2
2	Pliers	1
3	M4x12 Grounding Screw	1
4	Waterproof Self-adhesive Tape (to protect antenna connection part)	1
5	M6 Self-drilling Screw (to install the gateway on the wall)	4

8.2 Gateway Installation

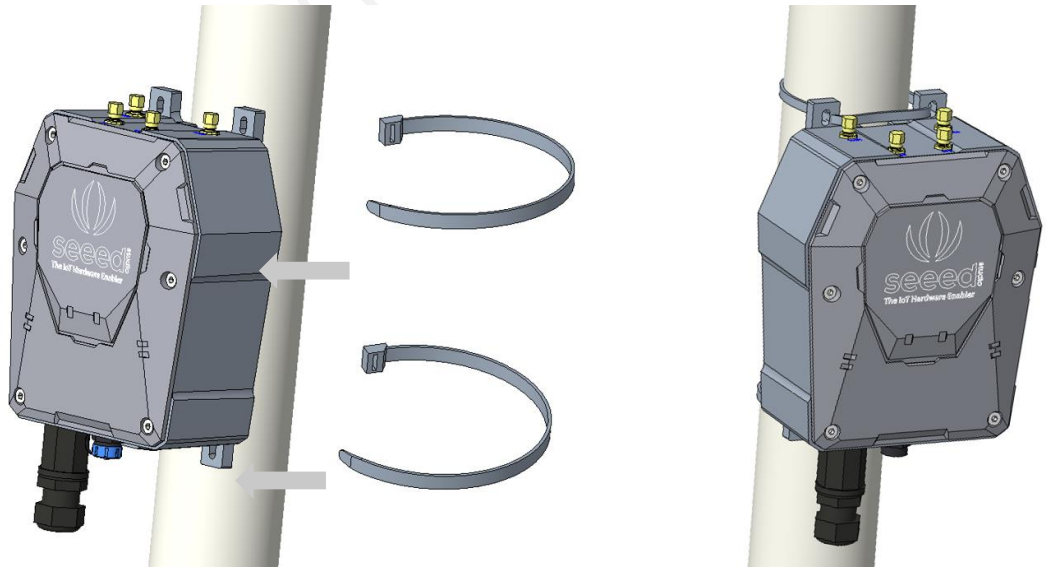
8.2.1 Gateway Installation Methods

- **Installing on a pole (Use the Mounts)**

Firstly, use M5 self-drilling screws (included in the package) to fasten the 4 brackets onto the gateway. And then use cable ties to fasten the gateway onto the pole. The recommended pole diameter is 70mm.

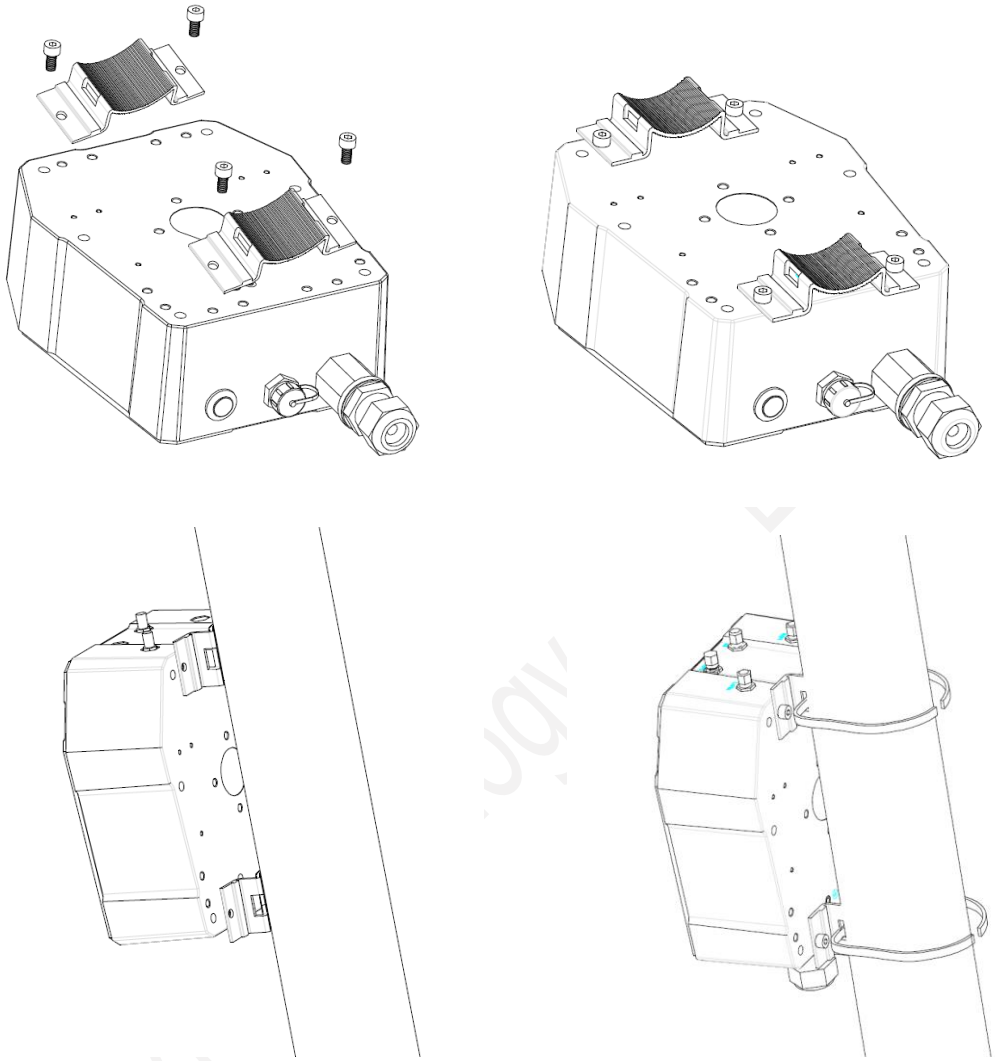


Put cable ties through the holes of the bracket and pull to fasten onto the pole. To get a better communication range, it is recommended to mount the gateway 3 meters above the ground. If there are tall buildings around, the gateway should be kept away from the building or mounted on top of the tall building.



- **Installing on a pole (Use the Ferrules and Aluminum pieces)**

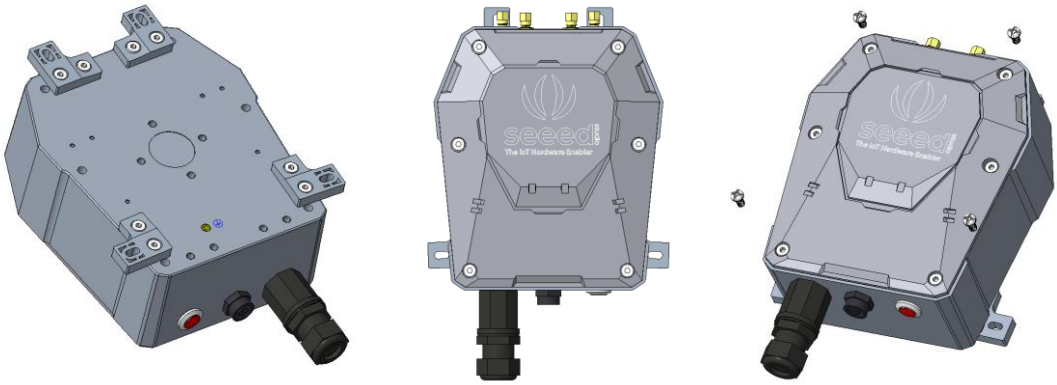
Firstly, use M5 self-drilling screws (included in the package) to fasten the 2 Aluminum pieces onto the gateway. And then use ferrules to fasten the gateway onto the pole. The recommended pole diameter is 76mm.



Note: If the pole is made of metal, the antenna should be pulled higher than the metallic part of the pole, or the communication signal will have interfered.

- **Installing on the Wall**

Firstly, use M5 self-drilling screws (included) to fasten the 4 brackets onto the enclosure of the gateway (refer to the image below for directions). And then fasten the gateway onto the wall with screws.

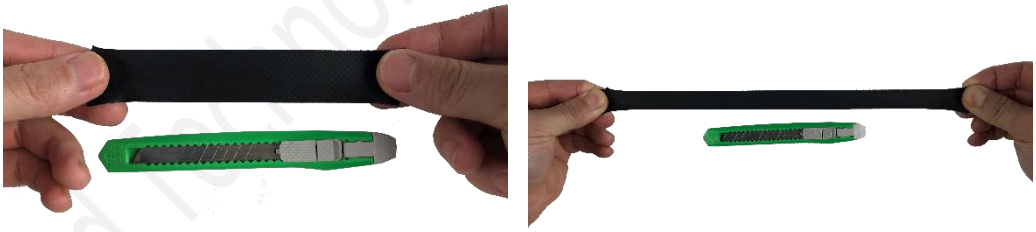


Note: The screws (that fasten gateway onto the wall) are not included in the package. Please prepare screws according to the wall materials (recommended screw diameter: 6mm).

8.2.2 Installation Precautions

1) In mountainous or thunderstorm-stricken areas, please take lightening protection measures. For the fiberglass LoRa antenna, you will need to install a lightening arrester and make sure it is connected to the ground. Besides, the gateway should be mounted lower than the lightening rod.

2) When installing the gateway in the outdoor environment, the connected part should be protected with waterproof tape, to enhance waterproof performance and lengthen device lifespan. As shown below, use self-adhesive tape to protect the connection. Take a rubber tape at the length of 10cm ~ 15cm, pull it to twice of that length



wind the tape clockwise to the connected part of the antenna.

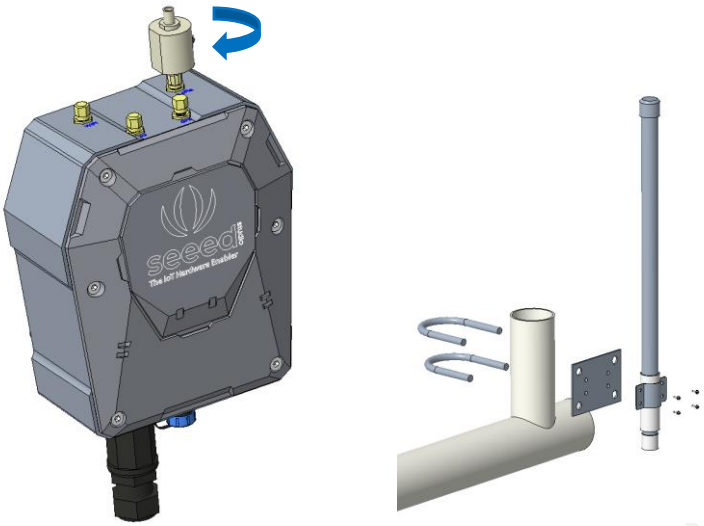
Note: The tape must be wound clockwise because the antenna is fastened clockwise. Otherwise, the antenna may loosen.

If the sensor has wires, install threaded tubes:

8.2.3 Installing Fiberglass LoRa Antenna

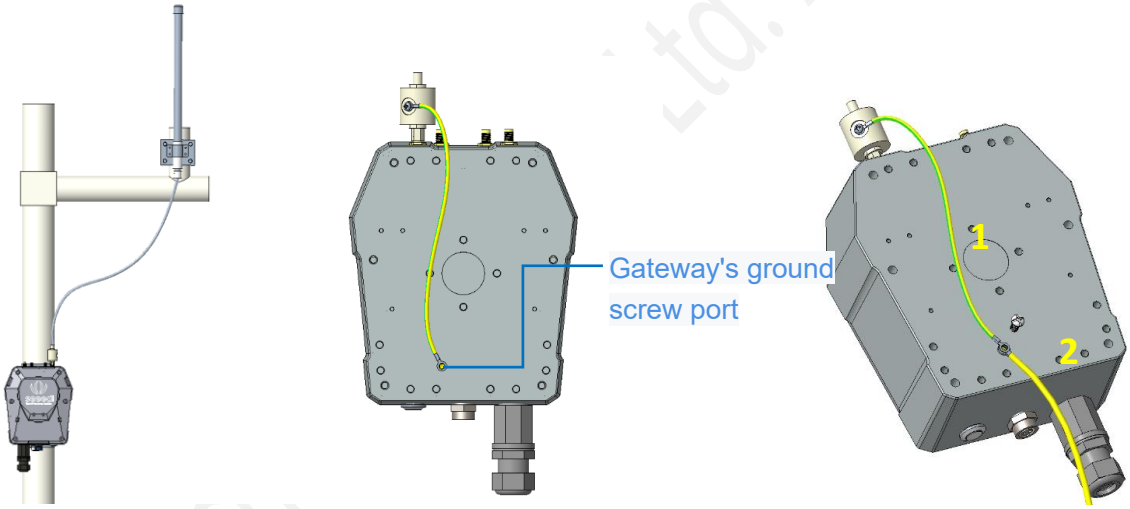
There are two kinds of LoRa antennas: the normal LoRa antenna (included in the package), and the fiberglass LoRa antenna (to be purchased separately). We will introduce how to install the fiberglass LoRa antenna.

1) Fasten the lightening arrester onto the antenna port.



2) As shown in the image below, please fasten the fiberglass antenna onto the base part, and then fasten the whole part onto the vertical cylinder (maximum cylinder diameter: 50mm).

3) Use a 1-meter antenna feed line to connect the lightening arrester with the fiberglass antenna.



8.2.4 Installing Ground Cable

Here we will connect the lightening arrester to the GND screw port on the gateway with a ground cable, and then connect the whole device to the ground. The image below shows the location of the GND port at the backside of the gateway.

1) Prepare two copper cables, a shorter one (approx. 30cm) for connecting the lightening arrester with the GND screw port (on the gateway), and a longer one for connecting the device to the ground.

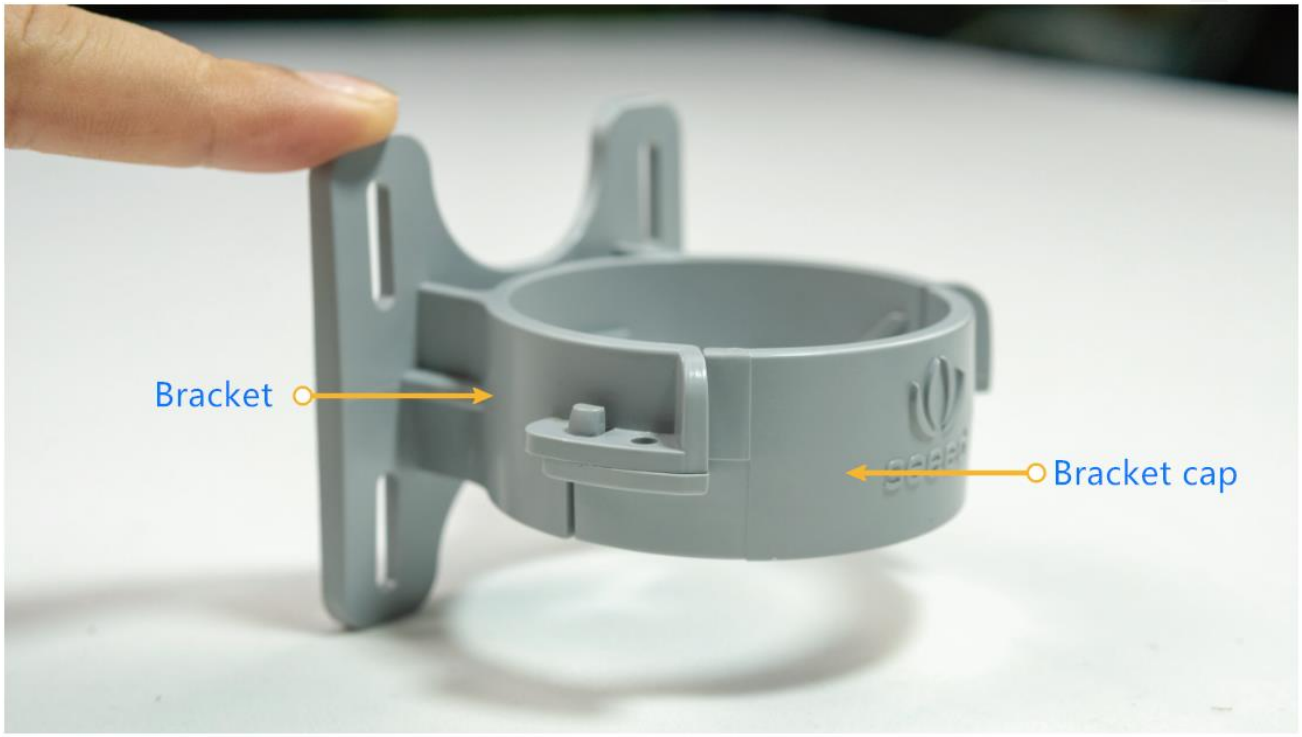
2) Fasten the lightening arrester to the short copper cable with screws, and then connect the two copper cables to the GND screw port. Use the screw to connect and fasten them.

3) Once the two cables are connected, connect the other end of the long cable to the ground. Depending on your actual installation environment, you can connect it to the ground directly or connect it to the copper ground bars.

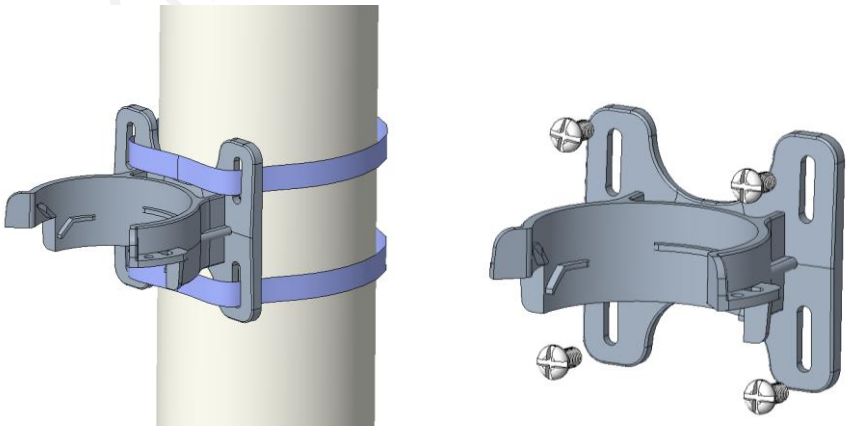
8.3 Installing Sensor Node

8.3.1 Installing the Sensor Node Bracket

Specially designed for installing SenseCAP Sensor Nodes, the bracket consists of a bracket and a sliding cap. With designated screw-holes, the bracket helps fasten the Sensor Node firmly onto a pole or a wall.



1) To install on a pole, you can use zip ties to fasten the bracket (recommended pole dimension is 50-70mm in diameter). Please refer to the following image for bracket directions.

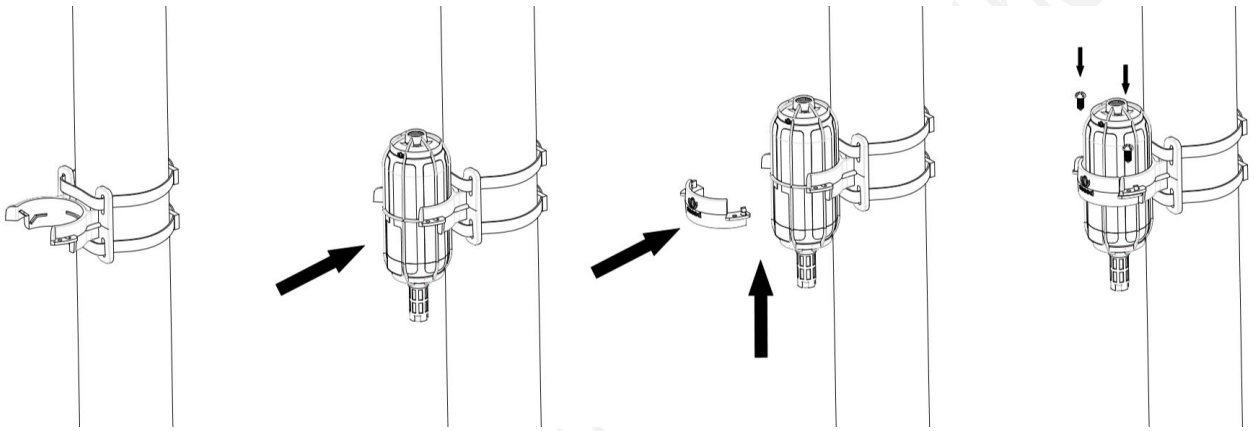


2) To install on the wall or other surfaces, you can use self-drilling screws to fasten the bracket onto the surface.

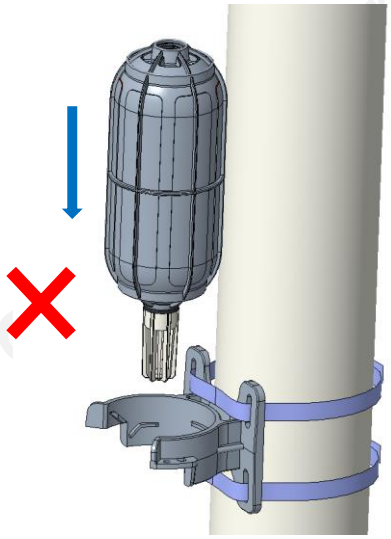
8.3.2 Installing Sensor Nodes

After installing brackets, let's install sensors.

1) The Sensor Probe should be placed vertically downward with the label facing outward. Be consistent with the bracket gap. Make sure the circle part in the middle of Sensor Node is aligned with the middle of the bracket, and then press the Sensor Node to fit into the bracket. A click/snap sound indicates that the Sensor Node has been installed successfully. Try to manually twist it to make sure the Sensor Node is locked to the bracket securely.
2) Secure by fastening the bracket cap as instructed in the image.
3) Place two self-drilling screws on the bracket to increase firmness and help prevent theft.



Note: Do not insert the Sensor Node into the bracket from the top, or it will not fasten the onto the bracket securely.

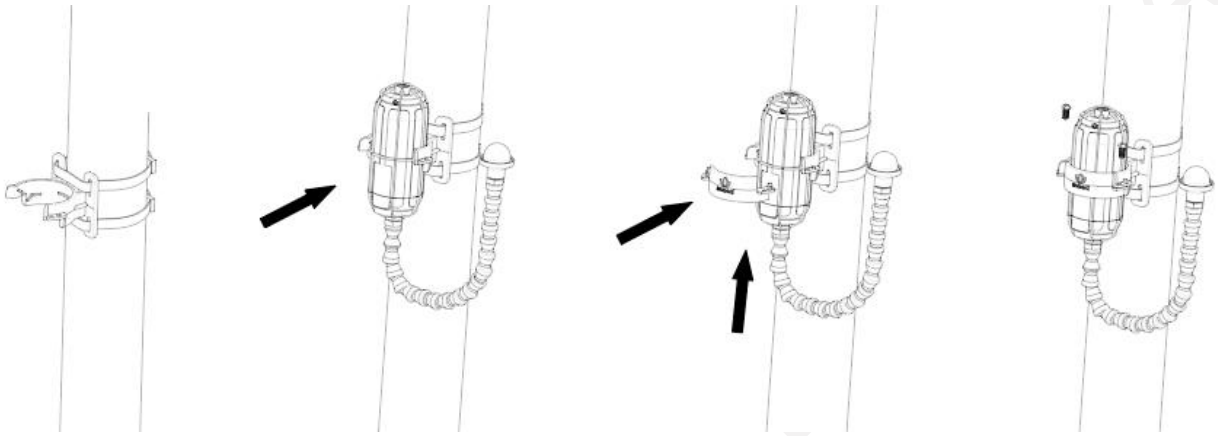


8.3.3 Dos and Don’ts in Installing Sensor Probes

The same instruction applies to installing the different Sensor Nodes. However, there are some tips to keep in mind when installing certain Sensor Nodes.

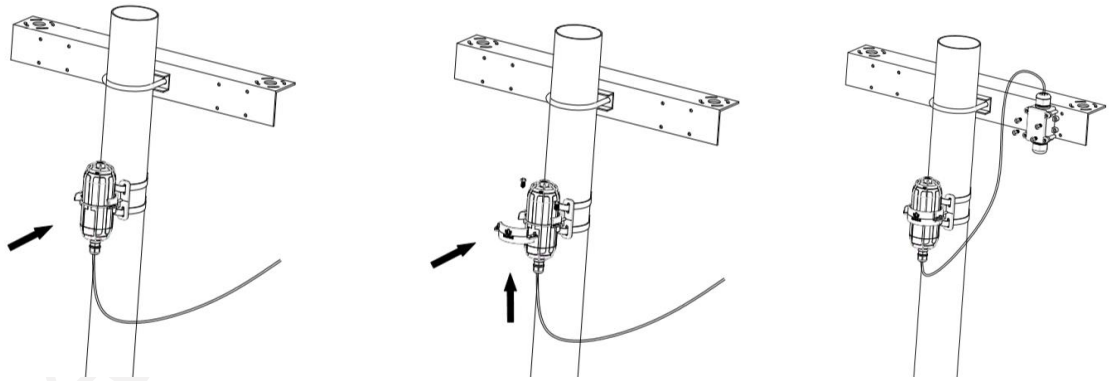
- **Light Sensor**

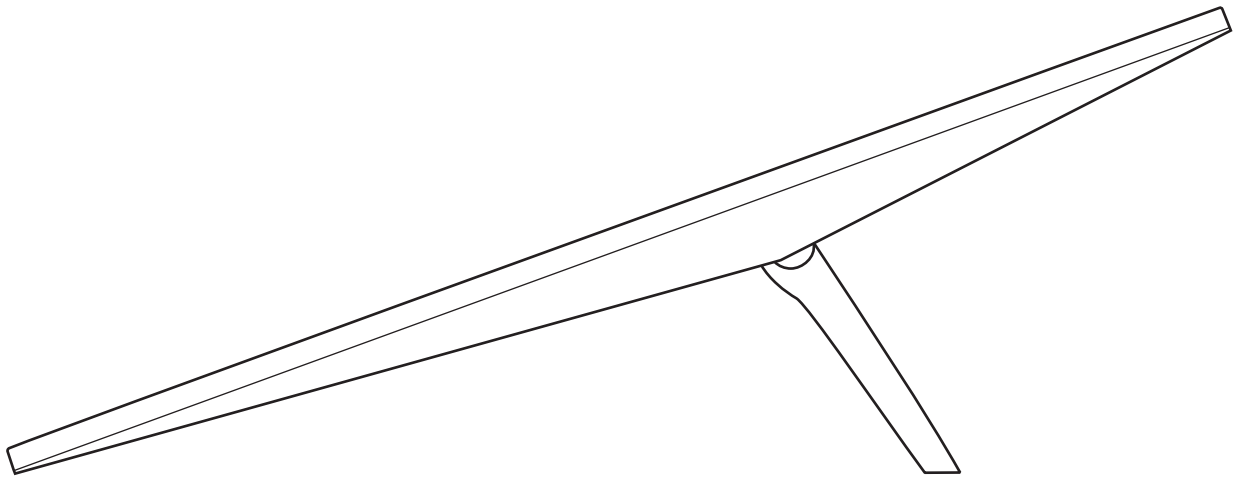
  The Sensor Probe of the Light Sensor needs to be placed vertically upward, and there should not be anything obstructing sunlight from the Sensor Probe.



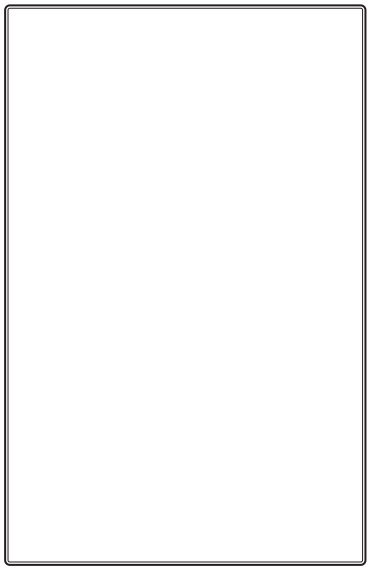
- **CO2 Sensor**

  The Sensor Probe can be fastened with self-drilling screws. Please refer to the image below for the probe direction. The end without the cables should point downward to prevent rain or dust from getting into the probe. Also, the device should be in a place with good ventilation.

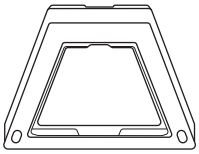




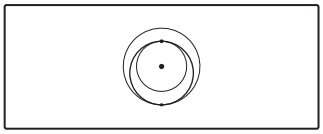
What's In The Box



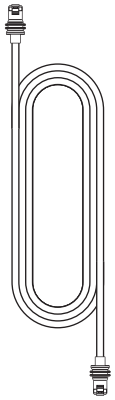
Standard



Kickstand

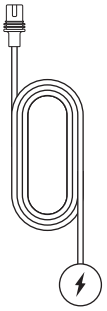


Gen 3 Router



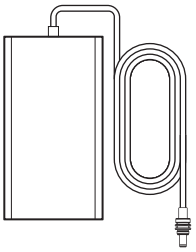
Starlink Cable

15 m
(49.2 ft)



AC Power Cable

1.5 m
(4.92 ft)

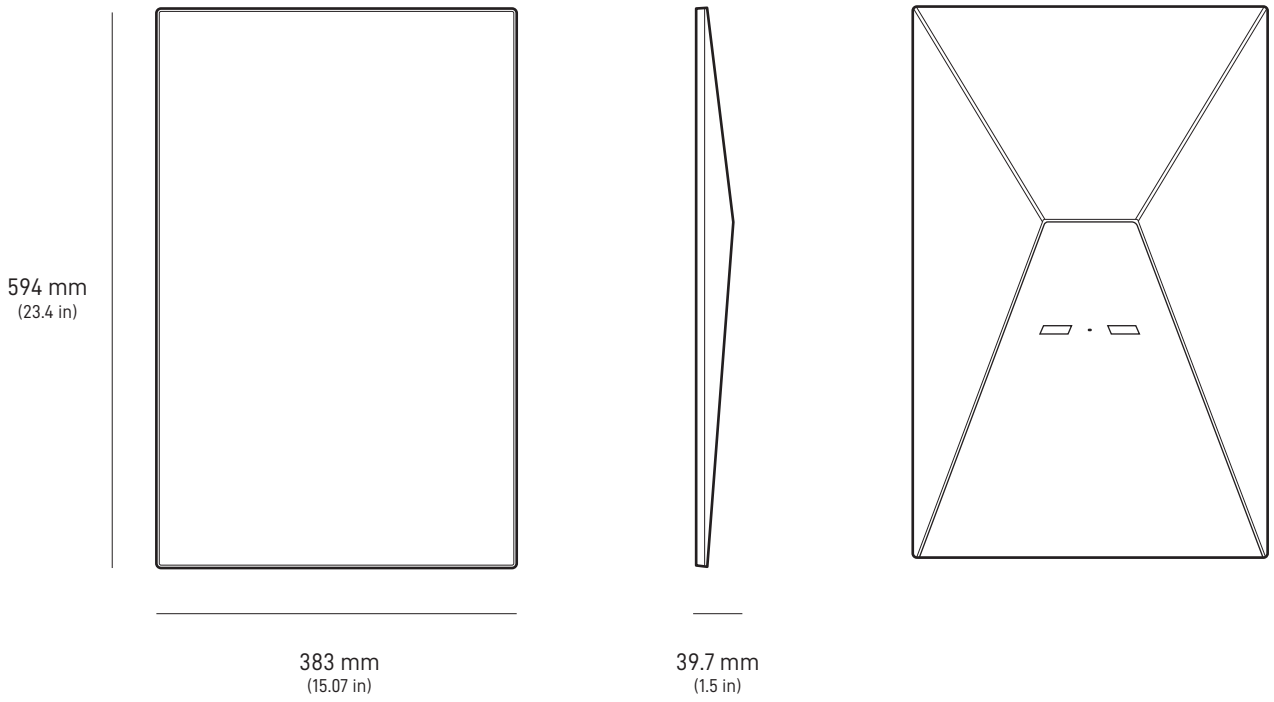


Power Supply

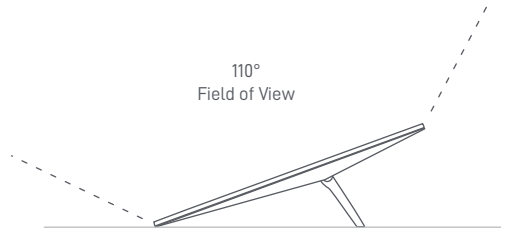
1.5 m
(4.92 ft)

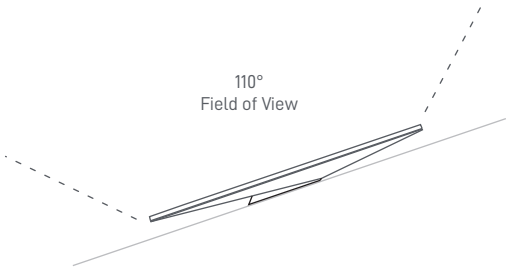
Package Weight: 6.73 kg (14.83 lbs)

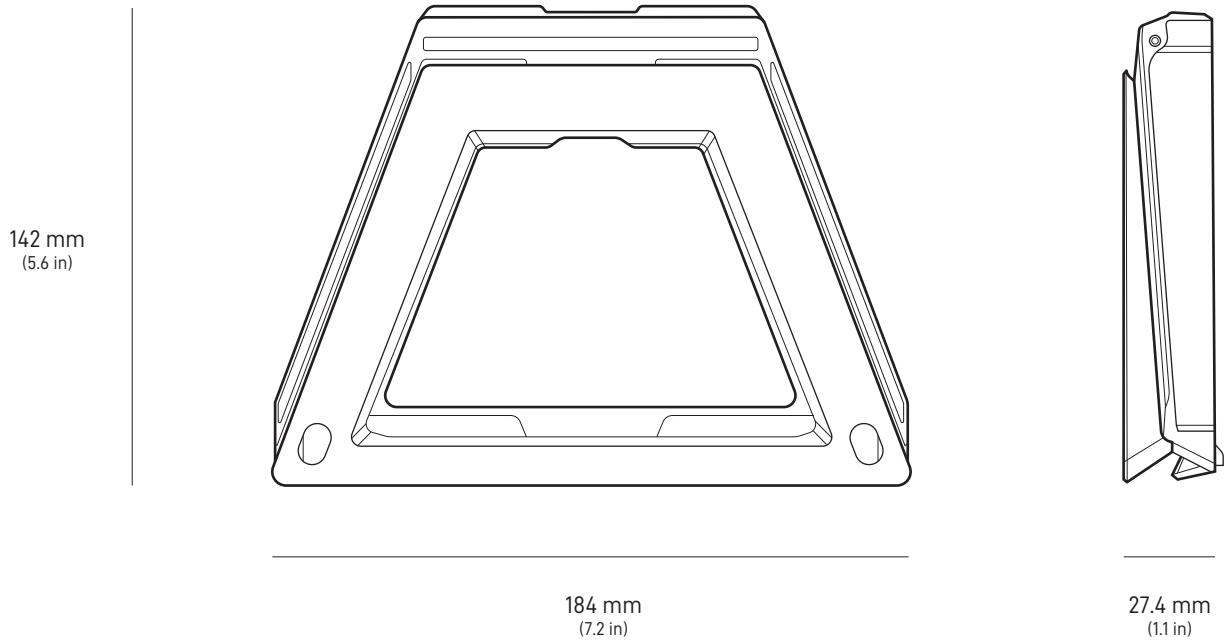
Package Dimensions: 652.4 x 451.7 x 97.9 mm (25.69 x 17.78 x 3.84 in)

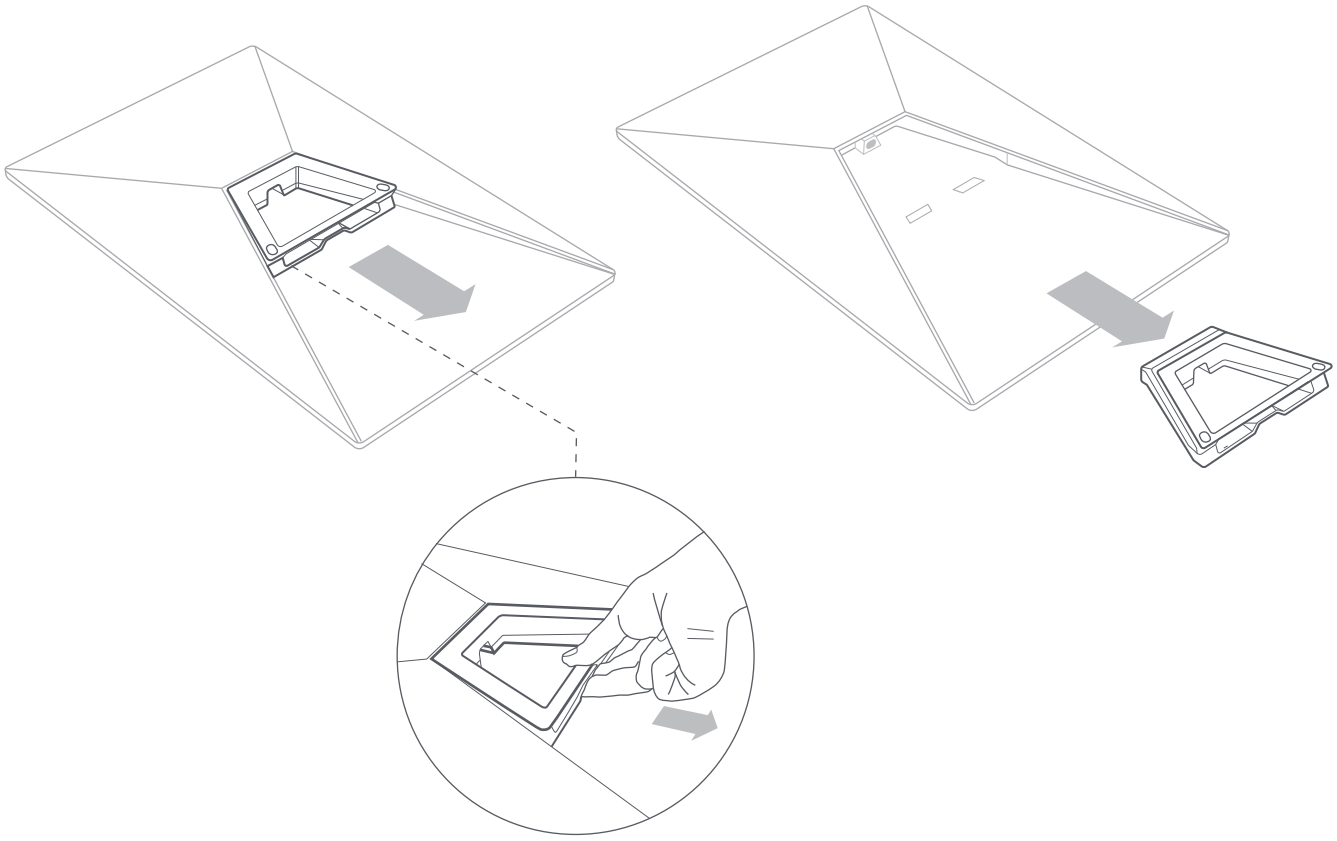


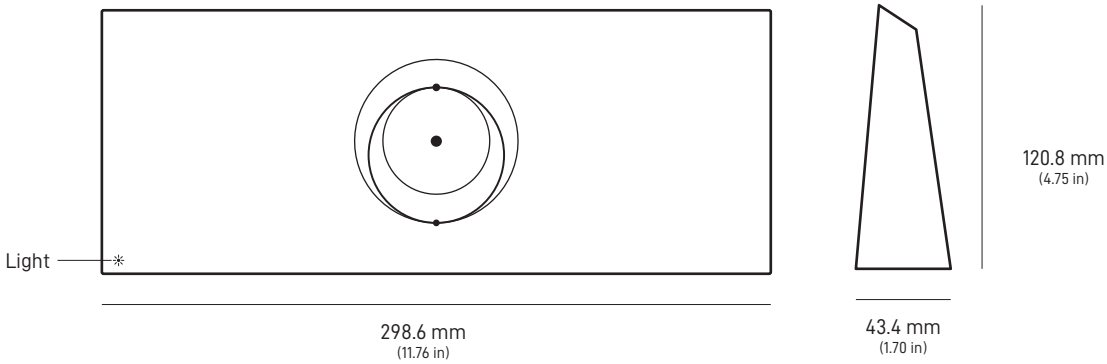
Antenna	Electronic Phased Array
Field of View	110°
Orientation	Software Assisted Manual Orienting
Weight	2.9 kg (6.4 lb) 3.2 kg (7 lb) with Kickstand 3.8 kg (8.3 lb) with Kickstand & 15 m Cable
Environmental Rating	IP67 Type 4
Operating Temperature	-30°C to 50°C (-22°F to 122°F)
Wind Speed	Operational: 96 kph+ (60 mph+)
Snow Melt Capability	Up to 40 mm / hour (1.5 in / hour)
Power Consumption	Average: 75 - 100 W

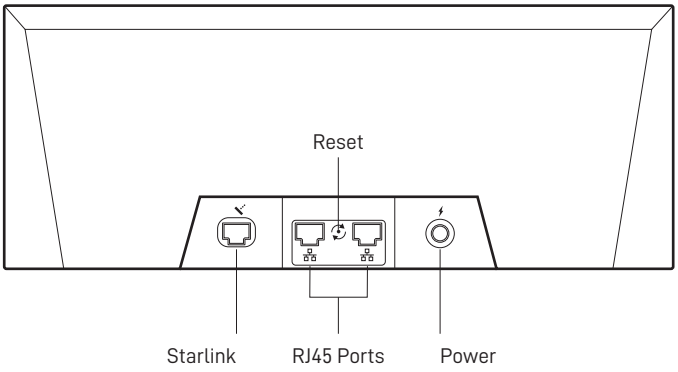




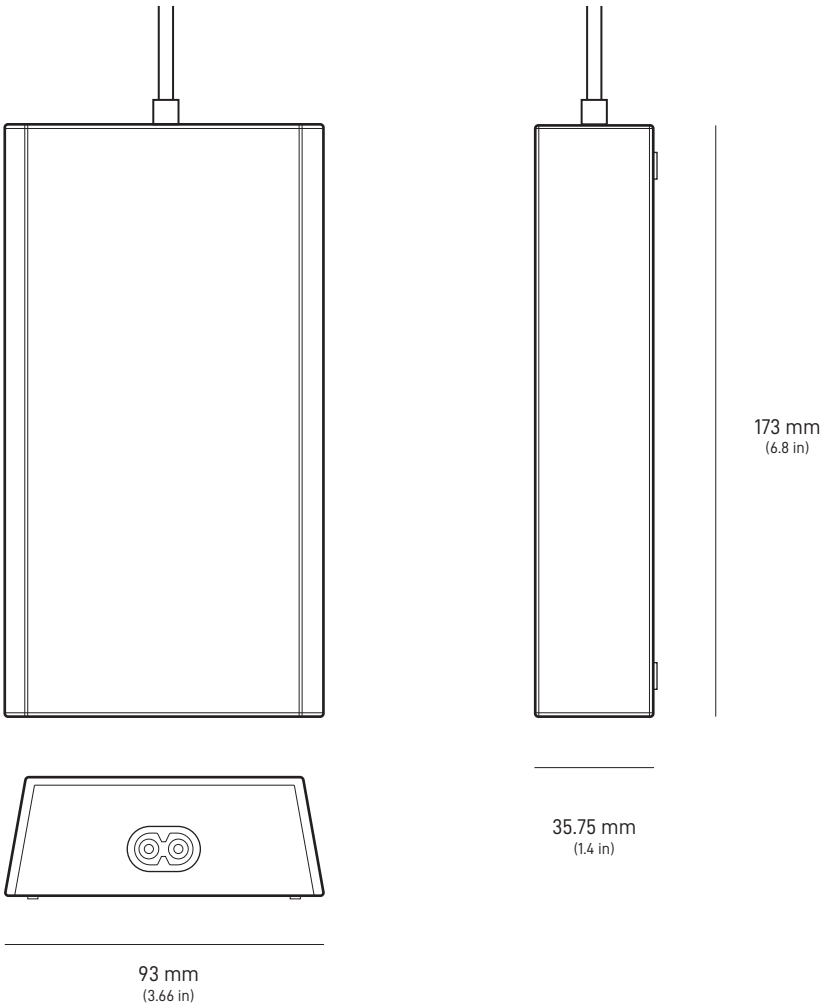

142 mm
(5.6 in)
184 mm
(7.2 in)
27.4 mm
(1.1 in)



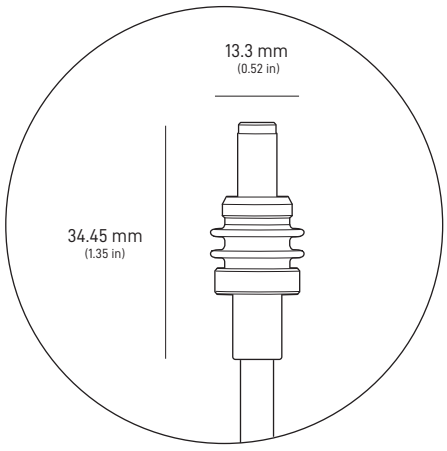


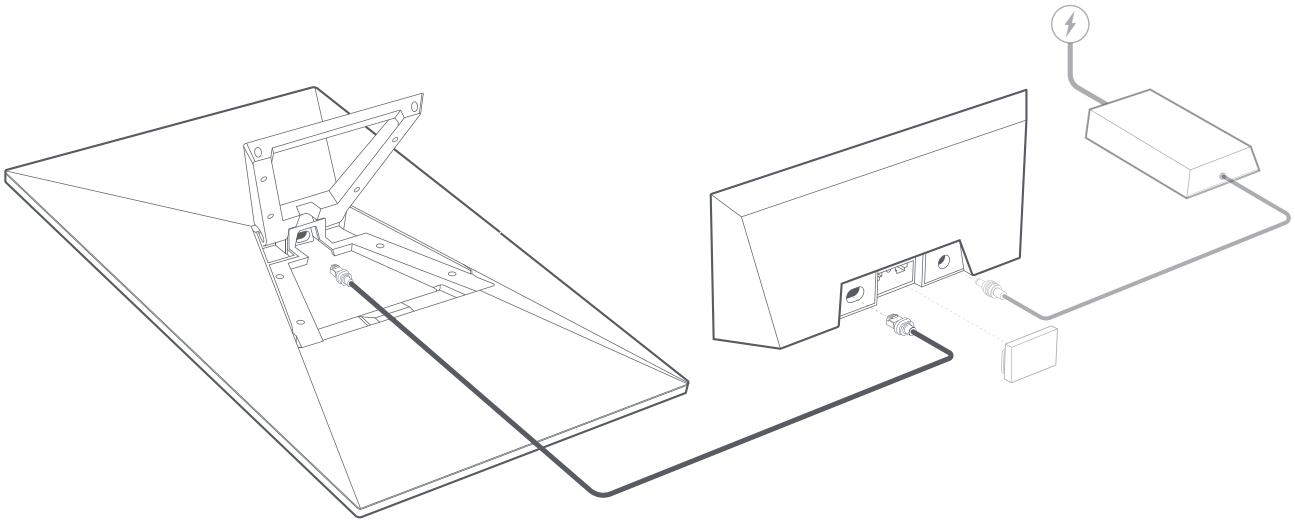


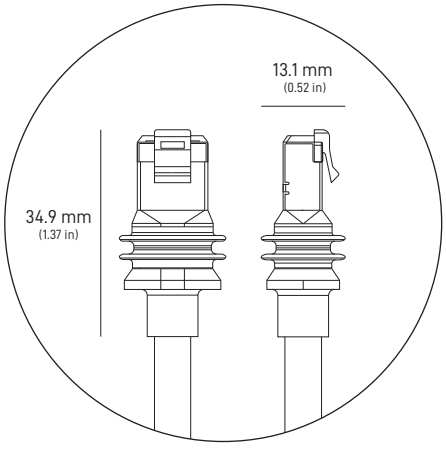
Product Dimensions	43.4 x 298.6 x 120.4 mm (1.7 in x 11.76 in x 4.74 in)
Wi-Fi Technology	802.11 a/b/g/n/ac/ax
Generation	WiFi 6
Radio	Tri Band 4 x 4 MU-MIMO
Ethernet Ports	Two (2) Latching Ethernet LAN ports with removable cover
Coverage	Up to 297 m^2 (3,200 ft^2)
Operating Temperature	-30°C to 50°C (-22°F to 122°F)
Weight	0.57 kg (1.25 lbs)
Security	WPA2
Environmental Rating	IP56 Rated (Water Resistant), configured for indoor use
Power Indicator	LED | face plate, lower left corner of router
Mesh Compatibility	Compatible with Starlink Gen 2 and Gen 3 Mesh Nodes, up to 3 Starlink Mesh Nodes *Not compatible with 3rd party mesh systems
Devices	Connect up to 235 devices



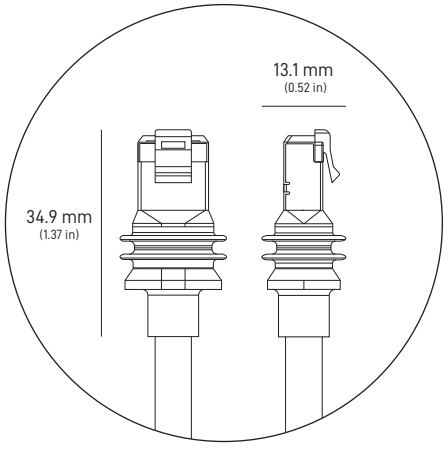
Product Dimensions	173 x 93 x 35.75 mm (6.8 in x 3.66 in x 1.4 in)
Weight	0.65 kg (1.43 lbs)
Environmental Rating	IP66 Type 4
Operating Temperature	-30°C to 60°C (-22 to 140°F)
Power Specifications	100-240V ~ 2.5A 50 - 60 Hz

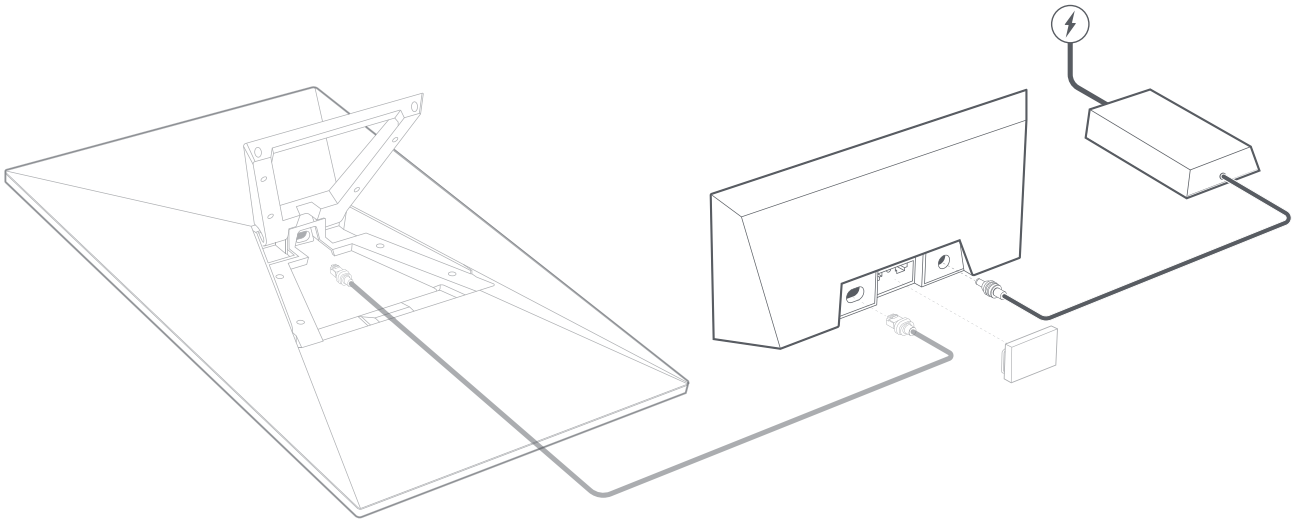


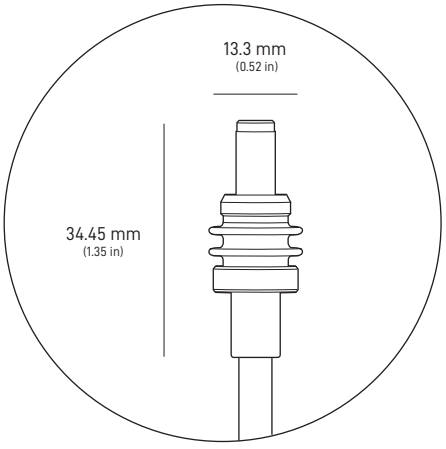




Pin 1	Wire Color	Pin 2
Shield	Drain - Bare Wire	Shield
8	Brown	8
7	White / Brown	7
6	Green	6
5	White / Blue	5
4	Blue	4
3	White / Green	3
2	Orange	2
1	White / Orange	1






13.3 mm
(0.52 in)
34.45 mm
(1.35 in)