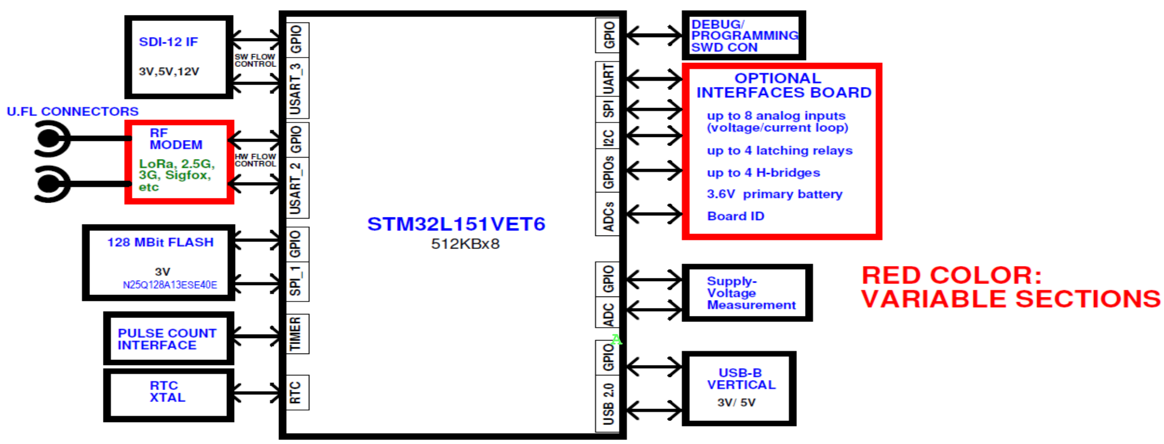
TBSL1 User Manual

Multi-sensors wireless bridge

The TBSL1 is a versatile outdoor Radio Telemetry Unit, acting as a multi-sensors wireless bridge. This is a flexible platform offering a rich connectivity for multiple usual sensors interfaces used in agricultural applications: SDI-12, pulses and analog. In addition, it provides support for interface boards expanding the capabilities of the platform (control of latch relays, additional analog sensors, digital interfaces: I2C and serial).

TBSL1 architecture allows support for various wireless modem technologies for communication with a remote server: LoRaWAN, GSM/WCDMA/LTE, Cat M1/Nb-IoT, Wi-Fi. Thanks to its large internal flash, TBSL1 can retain sensors measurements over a long period of time in case of communication outage with the server. The platform is fully configurable: multiple types of sensors can be attached, and each of them can operate on its own time basis for measurement.

Leveraging effective power management strategies, TBSL1 is also proposed with various power supplies variants (solar panel, internal/external batteries, …) boosting the autonomy of the platform in the fields.



TBSL1 module HW breakdown

Features

- Plug & Play modem board:
  - LoRaWAN 1.03 Class A
  - GSM/WCDMA/LTE (MQTT/HTTP/FTP)
  - Cat M1/NB2 (MQTT, HTTP, FTP)
  - Wi-Fi (MQTT)
- SDI-12 Standard V1.4
- Easy configuration with PC tool through USB port
- Ready for remote control and configuration options.
- Power Down Mode
- Power options: (battery/solar panel and external power supply)
- Embeds support for SDI-12, pulse and analog sensors; options for I2C and serial sensors.
- Configurable measurements and transmission intervals
- Internal data storage
- Rain gauge and flow meter applications
- Internal programmable alarms
- Outdoor (dust and water proof)
- Operating Temperature Range:
  -40°C - +85°C

Target Applications

- Any IoT application requiring the use of SDI-12, analog or pulse sensors.

Contents

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

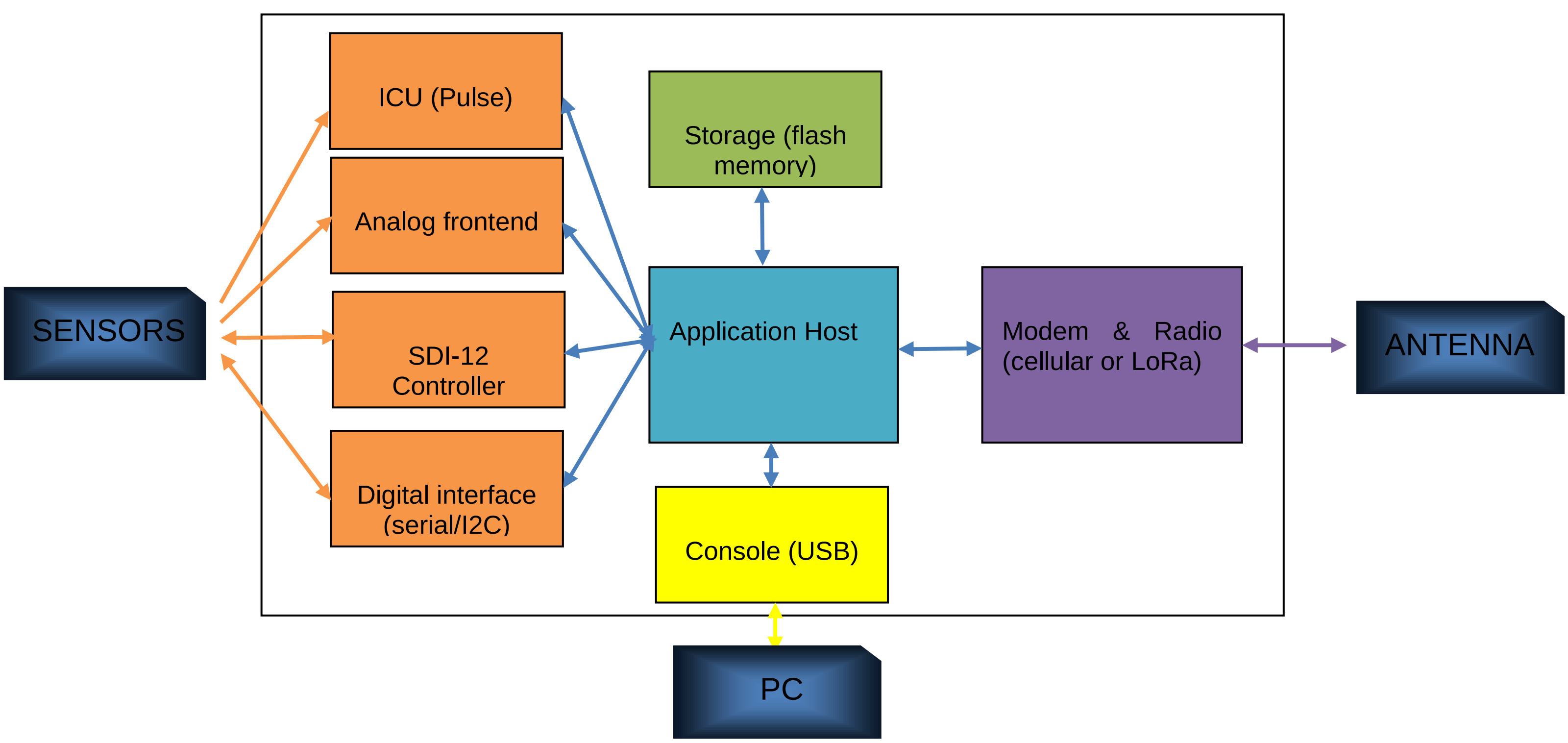
1 Platform overview

TBSL1 is the generic name of a range of multi-sensors wireless RTU, which are proposed with different modem options:

- TBSL1/RFB-LoRa: LoRaWAN Class A
- TBSL1/RFB-4GWW: LTE/WCDMA/GSM – MQTT v 3.1.1/HTTP/FTP communication protocols.
- TBSL1/RFB-NbIoT-CatM1: dual mode Nb-IoT/Cat M1 – MQTT v 3.1.1/HTTP/FTP communication protocols
- TBSL1/RFB-WiFi: Wi-Fi 802.11b/g/n – MQTT v 3.1.1 communication

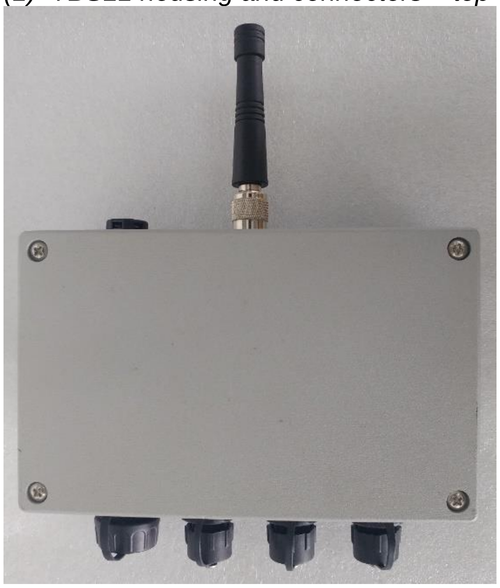
Main functional blocks:

- A SDI-12 controller, used to control TBSL1 SDI-12 sensors and send back measurements to internal host
- An input capture unit (ICU) to count number of pulse sensors ticks (eg rain gauge)
- An analog frontend to get measurements from analog sensors (2 channels)
- Digital interface block for I2C and serial sensors.
- A modem, used to transmit the measurements originating from the attached sensors to a remote application server (cellular modem) or to a LoRaWAN gateway (LoRa modem).
- An external flash to store sensors measurements.
- A USB connection to interface with a PC for configuration and console purposes.
- An application host, that controls:
  - Sensors commands and parameters configuration
  - Sensors measurements
  - Wireless transmission (cellular or LoRaWAN)
  - Sensors measurement and transmission intervals
  - Measurements storage
  - Power management



Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

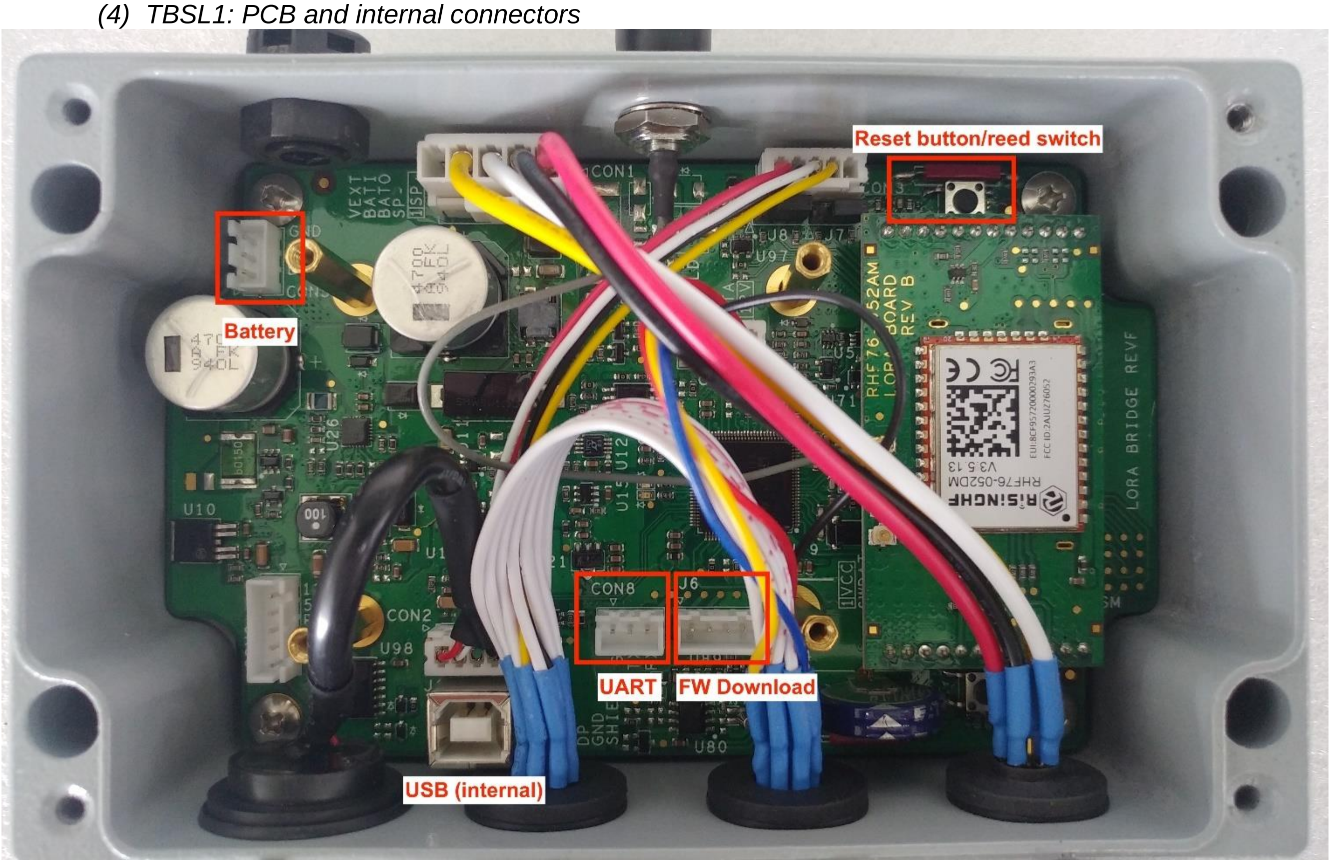
(1) TBSL1 housing and connectors – top view



(2) TBSL1: USB, sensors and solar panel connectors

(3) *TBSL1: antenna connector and vent*

(4) *TBSL1: PCB and internal connectors*



2 System overview

2.1 HW specification

2.1.1 Platform characteristics

Feature	Description	Additional information
	Sensors	
SDI-12 Analog Pulse	V1.4 0 – 2.5V / 4 – 20mA 0 – 3V / 1Hz max	Refer to Sensors support for further information.
	Modem	
Modem	LoRaWAN LTE/WCDMA/GSM Nb-IoT / Cat M1 Wi-Fi	Removable modem board. Refer to wireless modem section for further information.
	Platform	
Storage	Embedded 128Mbits flash memory	Amount of stored data depends on the number and kind of measurements taken on each reporting interval
Time syncronization	Precision RTC backed up by a super capacitor to retain the real time of the system	Time is retained even when the plaform is restarted or powered off
Platform reset	Internally: reset button Externally: reed switch	Not accessible for outdoor use Hover a magnet over reed switch
	Power supply (*)	
Power supply	TBSL1 with external solar panel and rechargeable battery	Outdoor use
	Solar panel	To be ordered separately
	Rechargeable battery	Not provided
	TBSL1 with external power supply (TBSL1-DC)	Not provided Indoor use only.
Current in idle mode	400 µA	With solar panel and battery
	Connectors	
Connectors	USB-B	Used for TBSL1 configuration (requires TBSL1 PC configuration tool)
	1 * 5 pins	LLT-M14BMF05 For solar panel connection
	2 * 7 pins	LLT-M14BMF07 For sensors connection
Antenna	Suitable for LoRaWAN, 2G, 3G and 4G communications Suitable for LoRaWAN 433MHz	Provided (868 – 915 MHz) Available on request

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

	Suitable for Wi-Fi modem	Provided (2.412-2.484 GHz)
Mechanical details		
Housing	IP67 aluminium housing: SP-AG-FA3-1 family, single piece gasket, IP67, TNC antenna connector, Gore-Tex vent, IP67 circular panel connectors	Suitable for outdoor use.
Operating temperature range	-40 to +85°C	

(*) Refer to Power supply chapter

2.1.2 Sensors support

Sensor	Support	Electrical characteristics	Additional information
SDI-12	Yes 2 ports available	12V, 1200 baud SDI-12 data interface with transient protection	SDI-12 v1.4 compliant More sensors can be connected by using a SDI-12 junction box (TBS04)
Analog	Yes 2 channels available	Voltage input: 0-2.5 V Current loop: 4-20 mA	Each channel configurable as either current loop or voltage input
Pulse	Yes 1 input available	0-3V Max frequency: 1Hz	2 multiplexed inputs are available. Only one can be used while the other must remain unconnected.
Modbus	No, but this can be achieved thanks to a SDI-12 to Modbus converter (TBS09)	Like SDI-12	Modbus RTU protocol

2.1.3 Wireless modem

Currently the following wireless technologies are supported by TBSL1:

- LoRaWAN Class A
- GSM/WCDMA/LTE
- NB-IoT/Cat M1
- Wi-Fi

Modem	Frequencies	Protocol	Additional information
LoRa	EU868 US915 US915HYBRID CN779 EU433 AU915 CN470 AS923 KR920 IN865	LoRaWAN 1.03 Class A LoRaWAN downlink	TBSL1 is configured by default to use 868-915 MHz band.

			Contact Tekbox sales for 433 MHz band support.
GSM/WCDMA/LTE	**GSM:** 850/900/1800/1900 MHz **WCDMA**: B1/B2/B4/B5/B6/B8/B19 **LTE-FDD**: B1/B2/B3/B4/B5/B7/B8/B12/B13/B18/B19/B20/B25/B26/B28 **LTE-TDD:** B38/B39/B40/B41	MQTT/HTTP/FTP	Global coverage
Nb-IoT/Cat M1	**Cat M1**: B1/B2/B3/B4/B5/B8/B12/B13/B18/B19/B20/B25/B26/B27/B28/B66/B85 **Cat NB2**: B1/B2/B3/B4/B5/B8/B12/B13/B18/B19/B20/B25/B28/B66/B71/B85	MQTT/HTTP/FTP	Global coverage
Wi-Fi	2.412~2.484GHz IEEE802.11b/g/n	MQTT	Downlink control on TBSL1-DC version only

2.1.4 Power supply options

There are two different TBSL1 variants with respect to power supply: one is powered by external solar panel and rechargeable battery (for outdoor use) or by external power supply (indoor).

External solar panel and rechargeable battery:

TBSL1: Solar panel powered	
Use	Outdoor (indoor: assuming the solar panel can receive enough sunlight)
Maximum delivered current	320 mA
External solar panel	
Max open load voltage	7 V
MPP voltage	5-6 V
Power rating	2-5 W (2W for sunny areas and 5W for less favorable areas)
Included	No

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

	(5W solar panel can be ordered from Tekbox)
Rechargeable battery	
Type	LiIon / LiPo single cell 3.7V
Capacity	1500-2500 mAh
HW	Must have a battery safety PCB and a 10kΩ NTC (3 wires interface: +, -, NTC)
Included	No
Charging temperature	0 – +50℃ *(Refer to Battery Charging Protection for further information)*

External power supply:

TBSL1: external DC power supply	
Use	Indoor / Outdoor (connection to the same waterproof 5 pins connector than the solar panel version)
Maximum delivered current	320 mA
External power supply	
DC voltage	6 – 18V
Connector	LLT-M14CM05 required to connect to TBSL1 5 pins connector.
Included	No

2.2 TBL1 connectivity

2.2.1 External connectivity details

TBSL1 provides following external connectivity:

- 2 connectors to connect SDI-12, analog and pulse sensors
- 1 connector to connect a solar panel
- 1 USB port for PC connection and battery charging
- 1 antenna connector

The following pictures show the external interfaces of TBSL1:

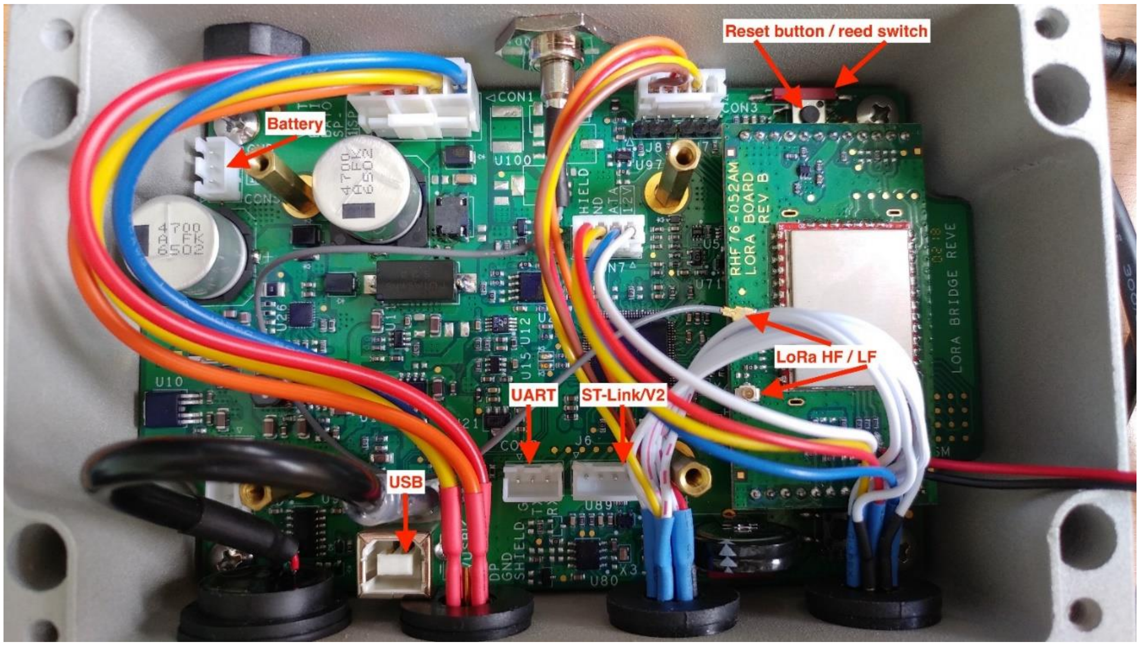
The pins allocation and description for each connector is as follows:

TBSL1: connectors description

2.2.2 Internal connectivity details

TBSL1 provides following internal connectivity:

- 1 ST-Link port for FW update and connection to an emulator (using ST-Link/V2 programmer)
- 1 battery connector
- 1 UART port to output diagnostic log
- 1 reset button
- 1 reed switch
- 1 internal USB connector sharing the same interface than the external USB



Notes:

- *3 pins battery connector Vcc/NTC/GND*
- *Reed switch allows to reset the unit when it is completely closed by applying a magnet on the external side of the housing.*
- *Internal USB connector is directly linked to external connector.*

2.3 Wireless communication

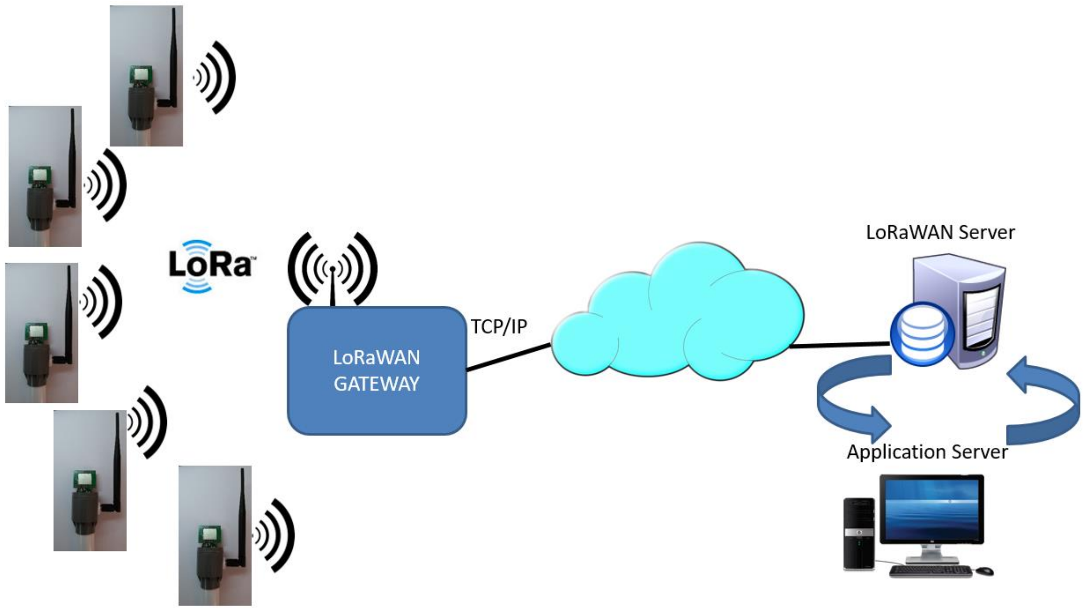
TBSL1 is designed with a interchangeable modem board socket which allows switching from one wireless technology to another by just changing the modem board while keeping the same main board HW.

2.3.1 TBSL1/RFB-LoRa platform

LoRaWAN modem board must be inserted into TBSL1 modem socket:

TBSL1/RFB-LoRa integrates within a typical LoRaWAN ecosystem: each TBSL1/RFB-LoRa is seen as a unique end node within a LoRaWAN private network (ie including at least one LoRaWAN gateway and a LoRaWAN server).

Following schematic shows a typical LoRaWAN deployment:



TBSL1/RFB-LoRa in a private LoRaWAN network

Before being used, TBSL1/RFB-LoRa must go through a configuration and provisioning phase to initialize system, SDI-12 and LoRaWAN parameters.

Device configuration is achieved by connecting TBSL1/RFB-LoRa to a PC and running the configuration tool provided along with this product. Further details pertaining to TBSL1/RFB-LoRa configuration are found in configuration chapter.

2.3.2 TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1 platforms

Cellular modem board must be inserted into TBSL1 modem socket:

LTE/WCDMA module:

Nb-IoT / Cat M1 module:

TBSL1 cellular versions (WCDMA/LTE dual mode and NB2/Cat M1 dual mode) provide 3 different methods to export the measurements to the application server:

- MQTT
- HTTP(S)

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

- FTP(S)

Only one export method can be active at the same time and it is enabled through TBSL1 configuration tool.

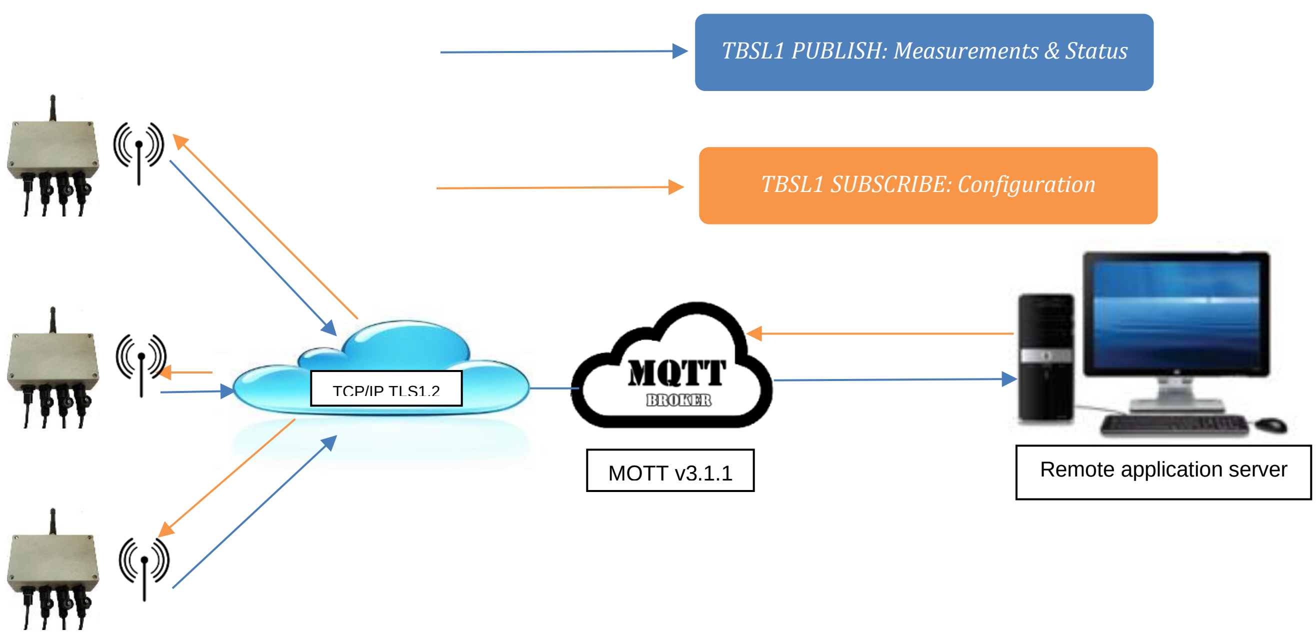
MQTT
This is the recommended TBSL1 export method.

Platform is fully compliant with MQTT v3.1.1 protocol, therefore any MQTT broker used to convey data to/from TBSL1 should ideally also be fully compliant with MQTT v3.1.1. This is especially important as the platform leverages some specific MQTT features that ensure its robustness and integrity:

- TLS1.2 encrypted communication (although plain and user/password protected communications are also supported but strongly not recommended). Provisionning is required to store TLS certificates and MQTT device identifier.
- Use of persistence to ensure data will be ultimately delivered even in case of temporary outage of the node (power or communication issue).
- Platform is designed to operate with QoS=1

Upon transmission interval the platform establishes a (secure) TCP/IP connection with the MQTT broker and publishes its measurements. At the same time the platform reacts to potential configuration messages it subscribed to.

Following schematic shows the interaction between TBSL1 and application server through MQTT:



HTTP(S)

TBSL1 implements HTTP(S) REST API POST method to export the measurements to a HTTP server.
Both IPv4 and Ipv6 are supported.

FTP(S)

TBSL1 implement FTP(S) protocol to export the measurements to a file. Both file name and path are configurable.
Both Ipv4 and Ipv6 are supported.

2.3.3 TBSL1-DC/RFB-Wifi platform

Wi-Fi modem board must be inserted into TBSL1 modem socket:

Wi-Fi module:

This modem variant exports the measurements only over MQTT. Furthermore TBSL1 downlink control over MQTT is only supported with the DC variant.

2.4 Product Features

The TBSL1 is a versatile platform designed to support various wireless modem and interface with a large set of sensors used in agricultural applications.

2.4.1 LoRaWAN features (TBSL1/RFB-LoRa only)

- LoRaWAN Class A
- Device activation: ABP / OTAA
- Pre-configured frequency plans (EU868, EU433, US915, US915HYBRID, CN470, CN779, AU915, AS923, KR920 and IN865) and configurable frequency plans
- Uplink communication – selectable confirmed or unconfirmed messages
- Downlink communication – for control of platforms parameters from a remote server
- Reconfigurable LoRaWAN parameters: identifiers (DevAddr, DevEUI, AppEUI) and security keys (NwkSKey, AppSKey, AppKey)
- Dual antenna connector: low (434MHz/470MHz) and high (868MHz/915MHz) bands
- LoRaWAN ACK and no ACK with REPEAT feature
- *Note: TBSL1/RFB-LoRa is fully configurable with respect to LoRaWAN parameters. Would you need a specific LoRaWAN configuration locked into TBSL1 please contact Tekbox sales.*

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

2.4.2 Cellular features (TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1)

- Frequencies bands:
  - Refer to Wireless Modem chapter for further details.
- TBSL1/RFB-4GWW triple mode modem (*):
  - GSM/WCDMA/LTE
  - 4 modes: Automatic (default), LTE, WCDMA, GSM
- TBSL1/RFB-NbIOT-CatM1 dual mode modem (**):
  - NB2 / Cat M1
  - 2 modes: NB2, Cat M1 (default)
- MQTT v3.1.1
  - Measurements reported to application server in JSON format.
  - Configurable QoS – QoS=1 recommended for optimum operations
  - Messages persistence (QoS=1 required)
  - Subscription to downlink topic to receive configuration commands
  - Security
    - User name and password
    - TLS 1.2 certificates
- HTTP(S)
  - Measurements reported to application server in JSON format through HTTP POST method.
  - Downlink configuration commands through HTTP GET method.
  - Security
    - User name and password
    - TLS 1.2 certificates
- FTP(S)
  - Measurements reported to application server as text file.
  - Downlink configuration through a text file containing the commands list
  - Security
    - User name and password
    - TLS 1.2 certificates

(*): applicable for the global LTE module. This may defer on custom modules available on request like specific network operator certified modules. Please contact Tekbox sales for further information.

(**): optional version with GSM fallback available upon request. Please contact Tekbox sales for further information.

2.4.3 Wi-Fi features (TBSL1-DC/RFB-Wifi)

- Frequencies bands:
  - Refer to Wireless Modem chapter for further details.
- Wi-Fi modem:
  - Wi-Fi networks scanning
  - Password protection

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

- MQTT v3.1.1
  - Measurements reported to application server in JSON format.
  - Configurable QoS – QoS=1 recommended for optimum operations
  - Subscription to downlink topic to receive configuration commands
  - Security
    - User name and password
    - TLS 1.2 certificates

2.4.4 System features

SDI-12 sensors	
Measurement commands	Up to 10 probes (virtual or physical sensor), each configured with a maximum of 10 SDI-12 measurement commands. Maximum of 100 measurement values on each measurement interval.
Data commands	Automatically handled by TBSL1
Measurements identification	Based on SensorId/Sub-sensorID indexes so the backend application can easily match a measurement with a specific sensor and physical value
Warm up delay	Yes, up to 300s
External power supply	TBLS1 can supply SDI-12 sensor with 12V *Note: for sensors that needs to be constantly powered.*
Analog	
Analog channels	2 Can be individually configured
Warm up delay	Yes, up to 300s
Parameters	Yes Scaling factor
Configuration	0-2.5V voltage input or 4-20mA current loop
Pulse	
Pulse channels	1
Configuration	selectable modes rain gauge or flow meter
Parameters	Yes Bucket capacity / Totaliser initial value
Alarms	

Sensors	SDI-12, analog and pulse 1 alarm per sensor max
Condition	Under/Over a limit Inside/Outside a range With/Without hysteresis
Reporting	Data packet to remote server
System timing	
Measurement interval	Up to 1440 minutes Configurable for each sensor
Transmission/reporting interval	Up to 1440 minutes
Transmission delay	Up to the transmission interval Used to stagger the transmission of several TBSL1 to limit collisions
Monitoring information reporting	Yes, battery level and RSSI reporting Configurable as a multiple of the transmission interval
TBSL1 configuration	
Remote configuration update	LoRaWAN: through LoRaWAN downlink packets Cellular: over MQTT/HTTP/FTP Wi-Fi: over MQTT
FW update	Over USB
Sensors and platform configuration	Over USB Through PC configuration tool
Operation modes	
Console	TBSL1 is in configuration mode, its system, sensors and modem parameters can be checked and updated. Moreover, it is possible to directly access and communicate with connected sensors and modem.
Logging	TBSL1 operates normally (logging and transmitting sensors measurements). In this mode it also logs into its internal storage diagnostic information that can be retrieved through the PC application when TBSL1 is connected with USB.

2.5 Technical references

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

2.5.1 SDI-12

SDI-12 is a standard for interfacing data recorders with microprocessor-based sensors. SDI-12 stands for serial/digital interface at 1200 baud. It can connect multiple sensors with a single data recorder on one cable. It supports up to 60 meters of cable between a sensor and a data logger.

The SDI-12 standard is prepared by

SDI-12 Support Group
(Technical Committee)
165 East 500 South
River Heights, Utah
435-752-4200
435-752-1691 (FAX)
http://www.sdi-12.org

The latest standard is version V1.4 and is available on the web site of the SDI-12 Support Group.

2.5.2 LoRaWAN

LoRaWAN is a MAC layer radio protocol for LoRa (technology owned by Semtech, www.semtech.com) developed and maintained by LoRa Alliance:

www.lora-alliance.org
LoRa™ Alliance
2400 Camino Ramon, #375
San Ramon, CA 94583
Phone: +1 925-275-6611
Fax: +1 925-275-6691

"**LoRaWAN™** is a Low Power Wide Area Network (LPWAN) specification intended for wireless battery operated Things in a regional, national or global network. LoRaWAN targets key requirements of Internet of Things such as secure bi-directional communication, mobility and localization services. The LoRaWAN specification provides seamless interoperability among smart Things without the need of complex local installations and gives back the freedom to the user, developer, businesses enabling the roll out of Internet of Things."

Further details available on LoRa Alliance website (https://www.lora-alliance.org/What-Is-LoRa/Technology).

2.5.3 MQTT

MQTT is an OASIS standard:

www.mqtt.org
www.oasis.org
OASIS
35 Corporate Drive Suite 150
Burlington, MA 01803-4238
USA

Phone: +1 781 425 5073
Fax: +1 781 425 5072
info@oasis-open.org

MQTT is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport. It is useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium. For example, it has been used in sensors communicating to a broker via satellite link, over occasional dial-up connections with healthcare providers, and in a range of home automation and small device scenarios. It is also ideal for mobile applications because of its small size, low power usage, minimised data packets, and efficient distribution of information to one or many receivers (more...)

3 Functional Description

3.1 Interface function

3.1.1 SDI-12

The SDI-12 standard defines a set of commands to configure sensors and to initiate measurements. Upon receiving specific commands, the sensor may carry out internal tasks, respond with information on conversion time or send measurement data.

SDI-12 commands are typically ASCII strings generated by the data recorder/controller firmware. The TBSL1 is connected to the TX output of the data recorder controller UART and converts the command strings to the logic levels and baud rate specified by the SDI-12 standard. Furthermore, the TBSL1 module handles breaks, marks and all other details of the SDI-12 protocol.

Upon receiving data or status information originated by a Sensor, the TBSL1 extracts the corresponding ASCII strings and presents it to the RX input of the data recorder controller UART.

The user has to configure TBSL1 with the set of SDI-12 commands that need to be executed. Only the measurement commands have to be configured as TBSL1 automatically handles the SDI-12 data commands.

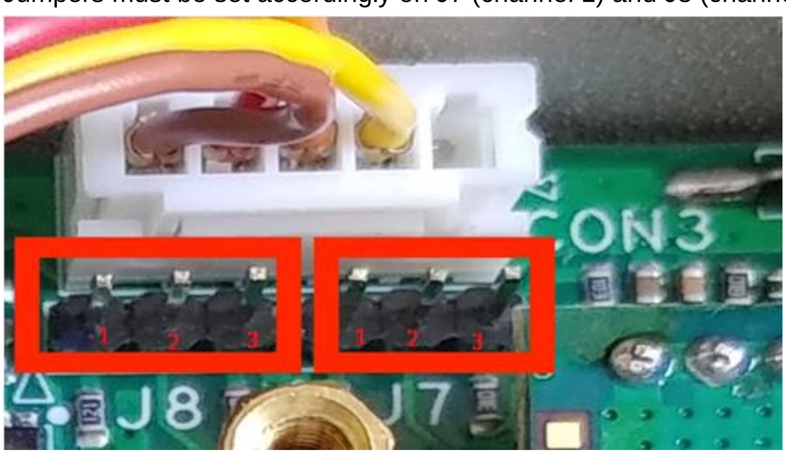
The PC configuration tool helps defining the 'sensor' ID and 'sub-sensor' ID which are indexes used to ease the integration into a backend application:

- 'Sensor' ID refers to a sensor with a specific SDI-12 address or analog channel
- 'Sub-sensor' ID refers to a specific SDI-12 command sent to a given SDI-12 sensor
- Notes:
  - For pulse sensor, both 'Sensor' and 'Sub-sensor' ID are set to zero
  - For analog sensor, the 'Sub-sensor' ID is systematically set to zero

3.1.2 Analog

TBSL1 provides 2 analog channels that can handle either voltage or current input.

Jumpers must be set accordingly on J7 (channel 1) and J8 (channel 0) for proper measurements:



Current input: jumper on J7/J8 1-2 pins; 4 – 20mA input.

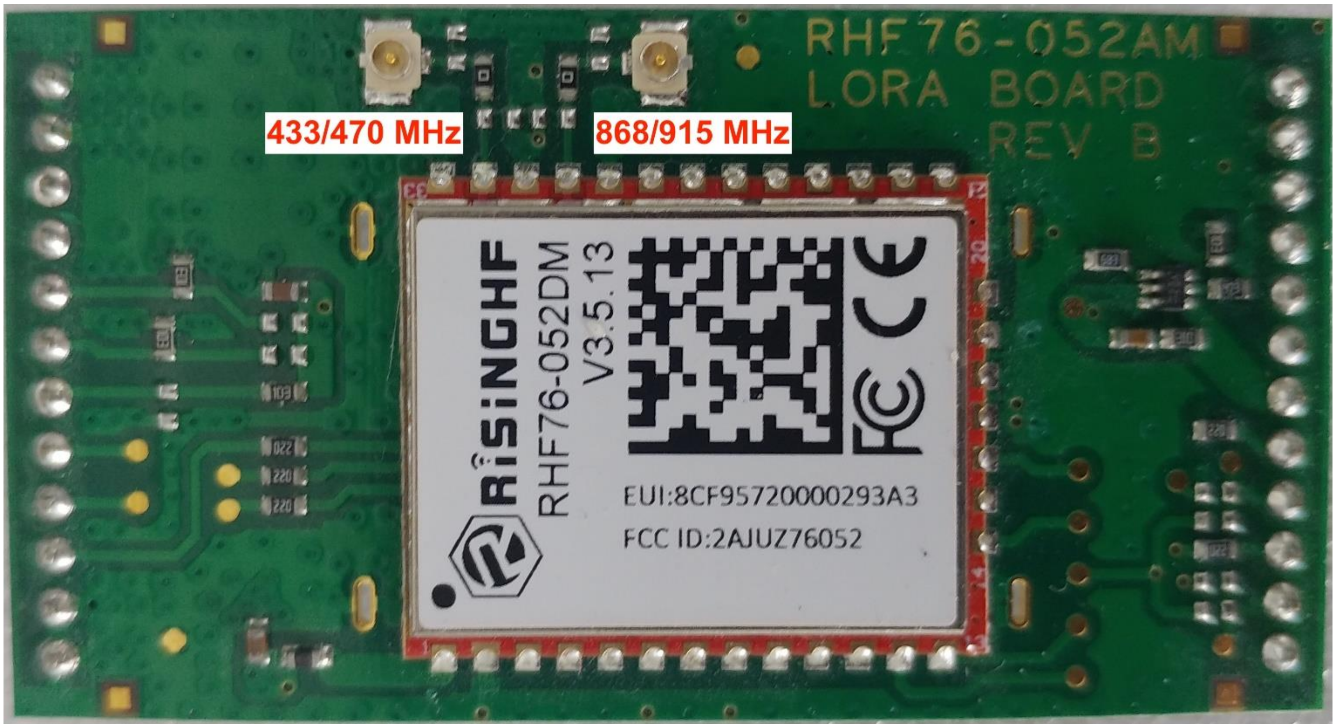
Voltage input: jumper on J7/J8 2-3 pins; 0 – 2.5V single ended input.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

3.1.3 LoRa modem and radio (TBSL1/RFB-LoRa)

TBSL1/RFB-LoRa embeds a LoRaWAN compatible modem and a dual band LoRa radio.

Depending on the frequency plan that is used (country specific), either low or high ISM band will be needed. It is therefore crucial to connect the antenna to the right connector.



Note: TBSL1 is delivered with 868/915 MHz compliant antenna by default. Please contact Tekbox sales in case a 433/470 MHz is needed instead.

As LoRaWAN uses free ISM bands, useable frequencies are subject to local regulation, and TBSL1/RFB-LoRa must be configured accordingly:

- To use supported LoRaWAN frequency plans for countries where this has been defined
- To use custom plans for other countries

Refer to latest LoRaWAN specification on www.lora-alliance.org

TBSL1/RFB-LoRa must be then used along with a LoRaWAN gateway that operates on the same frequency band: for example, if TBSL1/RFB-LoRa is deployed in Europe, both TBSL1/RFB-LoRa and gateway must be configured to use EU868 frequency plan.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

3.1.4 Cellular modem and radio (TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1)

When TBSL1 is fitted with cellular modem, the user needs first to select the radio technology that will be used and then choose the measurement export method.

Considering TBSL1 is used outdoor and can be deployed in areas where the cellular network provides limited coverage, it is advised to rely on MQTT for the preferred export method.

Usually MQTT communication is established over a secure TCP/IP data link relying on TLS1.2: this choice depends on the end user's requirements (vs plain communication). Therefore provisioning of the modem is required to store the client and CA certificates in this case (this also applies to HTTPS and FTPS methods).

This operation is normally handled by the distributor as physical acces to the modem chipset is required and involves several programming steps. The end user must then share beforehand details about its infrastructure so TLS certificates can be properly generated to allow the authentication and the connection to the MQTT broker.

It is critical that:

- Each modem communicating with a specific MQTT broker has its own and unique client certificate
- Each TBSL1 is configured with Tekbox PC configuration tool to set a unique MQTT Client ID

Additionally the MQTT communication can be established with a MQTT broker without any certificates with basic user name and password security.

Refer to the TBSL1 configuration tool user guide for further details.

3.1.5 Wi-Fi modem and radio (TBSL1/RFB-WIFI)

TBSL1 fitted with Wi-Fi modem board uses solely MQTT as export method and operates in this regard like the cellular version.

Nevertheless the Wi-Fi modem does not allow the use of MQTT persistent sessions, therefore only TBSL1 DC variant can support remote downlink control of the platform over MQTT (as the modem is constantly powered in this case unlike TBSL1 powered by solar panel).

3.2 System

TBSL1 is a versatile RTU (Remote Telemetry Unit): it is designed to trigger sensors measurements on a regular time basis, and transmit the measurements over LoRa or cellular modem to application server on defined time interval.

Each sensor operates on its own configurable time basis and measurements are stored in TBSL1 internal memory until they're transmitted.

3.2.1 Time intervals

Three different types of time intervals are defined in the system:

- Measurement interval
  - Period in minutes to perform sensors measurements and store data into internal memory.
  - Each sensor has its own measurement interval (except pulse sensors).
- Transmission interval

  - Period in minutes to transmit measurements stored in TBSL1 internal memory over the air
- Monitoring interval
  - Period in minutes to transmit monitoring information (eg battery level) over the air to the application server. This must be a multiple of the Transmission Interval.

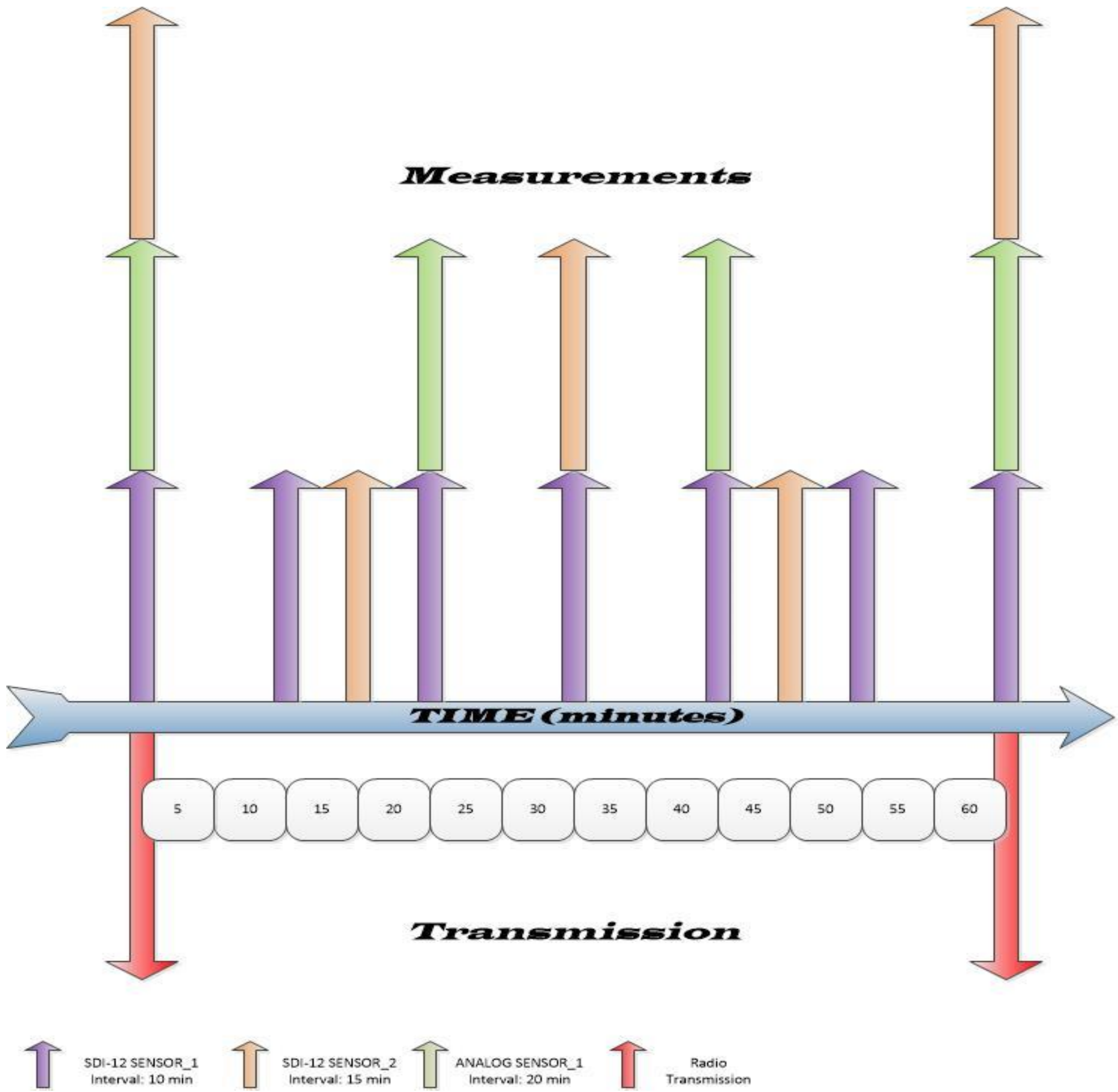
These parameters are configured with the TBSL1 Configuration Tool.

Intervals are set in minutes from 1min to 1440 minutes (ie 24 hours) and must satisfy the following condition:

1440 modulo (Interval) = 0

Following diagram shows how TBSL1 operates with 3 sensors having their own measurement interval (10, 15 and 20 minutes) and 1 hour transmission interval:

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com



3.2.2 Communication outage

TBSL1/RFB-LoRa

TBSL1/RFB-LoRa platform is able to handle 2 different situations related to communication outage:

(1) Some stored measurements have not been transmitted because of low battery preventing radio transmission or battery ran flat.
(2) Data is not received by the server because of radio transmission issue, gateway is down, LoRaWAN server outage, etc…

Situation (1) is automatically handled by TBSL1/RFB-LoRa: when radio transmission resumes it will transmit the measurements that have not been previously sent.

Situation (2) is handled by TBSL1/RFB-LoRa only if the device has been configured to use confirmed messages as per LoRaWAN standard: in this case, whenever a packet of data is sent by TBSL1/RFB-LoRa, the LoRaWAN server must send back an acknowledge to confirm the packet has been successfully received.

Would this acknowledge not be received, transmission is aborted. The packet will be retransmitted upon the next transmission interval, along with subsequent data packets stored in internal memory.

Alternatively, the use of LoRaWAN repeat feature for unconfirmed messages can increase the successful reception of packets (messages are duplicated as sent several times by LoRaWAN modem, reducing then the packets loss rate). It is recommended to set the repeat counter to 3 for unconfirmed messages (nonetheless, this tuning is dependant on the deployment configuration, especially the distance between the node and the gateway and the topography).

TBSL1 embeds a large 16Mb external flash memory that is solely used to store sensors measurements.

Measurements are overwritten in flash only when it becomes full: depending on the programmed measurement and transmission intervals, and the volume of data to be stored (related to the number and types of connected sensors), it is possible to assess the maximum outage time without loosing data (cf Data format).

TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1

Both platforms are able to handle 2 different situations related to communication outage:

(1) Some stored measurements have not been transmitted because of low battery preventing radio transmission or battery ran flat.
(2) Data is not received by the server because of radio transmission issue

Situation (1) is automatically handled by TBSL1: when radio transmission resumes it will transmit the measurements that have not been previously sent.

Situation (2) is handled through persistence feature of MQTT protocol and QoS=1. This ensures delivery of messages even in case of communication outage between platform/application server and the MQTT broker. This is why using MQTT as export protocol is the preferred method.

3.2.3 Uplink data format (TBSL1/RFB-LoRa)

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

This section describes the payload format for uplink messages sent by TBSL1/RFB-LoRa.
Several reporting messages are transmitted by TBSL1:

- Common data message
- Pulses report message
- Analog report message
- SDI-12 report message
- Battery report message

All packets can be collected, parsed, stored in an internal database and made available to external data analytics platforms through Tekbox IoT Broker. For further information, contact your local Tekbox distributor.

Alternatively, the end user can parse the received packets from the LoRaWAN server by using the following API to analyze the payload format.

Following identifiers are used within reporting messages:

Identification for the Report Types

No	Report Types	ID *(character)*	Descriptions
1	ECP_COMMON_REPORT_ID	'C'	This type will report common information in node device (TBSL1)
2	ECP_PARAM_REPORT_ID	'P'	This type will report parameters of all sensors that embed in the TBSL1.

Definition for Power Supply Identification

No	Power Supply ID	Value *(character)*	Descriptions
1	ECP_SUPPLY_SOLAR_USB_ LIPO_ID	'0'	Solar panel or USB charging a LiPo battery
2	ECP_SUPPLY_SOLAR_USB_ NIMH_ID	'1'	Solar Panel or USB charging a NiMh pack
3	ECP_SUPPLY_EXT_6TO12V_ID	'2'	External 6..12V power supply, no internal battery
4	ECP_SUPPLY_INTERNAL_LI_BAT_ID	'3'	Internal primary Li battery, 3.6V.

Definition for Sensors Identification

No	Sensor Type ID	ID *(character)*	Descriptions
1	ECP_SDI12_ID	'S'	Identifier for SDI-12 sensors
2	ECP_PULSE_CNT_ID	'P'	Identifier for Pulse Count sensor
3	ECP_ANALOG_ID	'A'	Identifier for Analog sensors
4	ECP_EXT_VOL_ID	'E'	Identifier for external voltage supply
5	ECP_BAT_VOL_ID	'B'	Identifier for battery voltage
6	ECP_RSSI_ID	'R'	Identifier for LoRaWAN "Received Signal Strength Indicator" (RSSI)
7	*ECP_SIM_CARD_ID*	*RFU*	*RFU*
8	*ECP_CAMERA_ID*	*RFU*	*RFU*

9	*ECP_GPS_SENSOR_ID*	*RFU*	*RFU*
10	*ECP_DIGITAL_ID*	*RFU*	*RFU*

- **Common data message**

Message for Common Data Report								
Report ID	Timestamp	Device ID	FW Version	Power Supply ID	Sensor Number	Board Status ID	Space character	RSSI
1 byte	*17 bytes*	*8 bytes*	*8 bytes*	*1 byte*	*1 byte*	*1 byte*	*1 byte*	*RSSI format*

Message's fields:

- Report ID
  - ✓ Description: character 'C' that is the first character of the common report message.
  - ✓ Field size: 1 byte
- Timestamp
  - ✓ Format: YY:MM:DD:hh:mm:ss (24h format)
    - YY: 2 characters representing 2 digits year's number.
    - MM: 2 characters representing 2 digits month's number.
    - DD: 2 characters representing 2 digits date's number.
    - hh: 2 characters representing 2 digits hours number.
    - mm: 2 characters representing 2 digits minutes number.
    - ss: 2 characters representing 2 digits seconds number.
  - ✓ For example: the time stamp October 27, 2016, 13 hours 45 minutes and 00 seconds is encoded as "16:10:27:13:45:00"
- Device ID or Node ID
  - ✓ Description: 8 characters identifier.
  - ✓ Field size: 8 bytes
  - ✓ Example: "abcd1234"
- FW Version
  - ✓ Description: firmware version coded over 8 characters.
  - ✓ Field size: 8 bytes
  - ✓ Example: version 04.00.01.0b is encoded as "0400010b".
- Power Supply ID
  - ✓ Description: refer to the table "Definition for Power Supply Identification".
  - ✓ Field size: 1 byte
  - ✓ **Not supported by HW, reserved for future use.**
- Sensors Number
  - ✓ Description: one character that is the number of all sensors connected to the platform. The value range is from '0' to '9' and from 'a' to 'z'.
  - ✓ Field size: 1 byte
  - ✓ Example: Number 15 is encoded as "e".
- Board status ID:
  - ✓ Description: this field indicates if the board has just startup or is running.
  - ✓ Field size: 1 byte
  - ✓ Example: 'R' (Running) or 'S' (Startup).
- Space character
  - ✓ This field is space character that delimits parameters between the board status ID and RSSI.

- ✓ Field size: 1 byte
- ➢ RSSI
  - ✓ Value format of the RSSI: pd
    - o p – the polarity sign (+ or -)
    - o d – numeric digits

    The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1.
  - ✓ Field size: changeable
  - ✓ For example: value of RSSI is -45 that is encoder as "-45".

Example of message for Common Data Report

- ➢ Fields:
  - ✓ Report ID: Message for common data reporting. Value of this field is character 'C'.
  - ✓ Timestamp: February 25, 2019, 17 hours 30 minutes and 00 seconds is encoded as "19:02:25:17:30:00"
  - ✓ Device ID: Device ID is 00004AC1 and is encoded as "00004AC1"
  - ✓ FW Version: The firmware version is 04.00.01.0b and is encoded as "0400010b".
  - ✓ Power Supply ID: Power supply is a LiPo battery charged by solar panel or USB, encoded as character '0' (ECP_SUPPLY_SOLAR_USB_ LIPO_ID).
  - ✓ Sensor Number: total number of connected sensors (ie SDI-12 + analog + pulse count sensors) that is 15. This value is encoded as "f".
  - ✓ Board status ID: 'R' board is running
  - ✓ RSSI: value of RSSI is -53 and is encoded as "-53"
- ➢ So the message for common data report is as below:

  C19:02:25:17:30:0000004AC10400010b0fR -53

- **Pulse sensor report**

Message for Pulse Count Sensors Report									
Report ID	Sensor Type ID	Timestamp	SensorID	SubSensorID	Number of Parameters	Space Character	Parameter 1	Space Character	Parameter 2
1 byte	*1 byte*	*17 bytes*	*1 byte*	*1 byte*	*1 byte*	*1 byte*	*8 bytes*	*1 byte*	*12 bytes*

Message's fields:

- ➢ Report ID
  - ✓ Value of this field: character 'P' that is first character of the parameter report message.
  - ✓ Field size: 1 byte
- ➢ Sensor Type ID
  - ✓ Format: cf table "Definition for Sensor Identification". Sensor's Type ID for this message is character 'P'.
  - ✓ Field size: 1 byte
- ➢ Timestamp
  - ✓ Format: YY:MM:DD:hh:mm:ss (24h format)
    - o YY: 2 characters representing 2 digits year's number.
    - o MM: 2 characters representing 2 digits month's number.
    - o DD: 2 characters representing 2 digits date's number.
    - o hh: 2 characters representing 2 digits hours number.
    - o mm: 2 characters representing 2 digits minutes number.

- ss: 2 characters representing 2 digits seconds number.
- ✓ For example: the time stamp October 27, 2016, 13 hours 45 minutes and 00 seconds is encoded as "16:10:27:13:45:00"

➢ SensorID
- ✓ This field is the virtual ID of the pulse sensor attached to TBSL1. Value range can be '0', '1', '2', '3', '4', '5', '6', '7', '8' and '9'. (Note: TBSL1 HW rev F supports only 1 pulse input hence SensorID='0').
- ✓ Format: decimal
- ✓ For example:
  - Only one pulse sensor connected to TBSL1, its index is 0.

➢ SubSensorID:
- ✓ ID reflecting the pulse application.
  Format: '0' for rain gauge and '1' for flow meter.

➢ Number of parameters:
- ✓ Number of returned parameters
  Format: '2' (2 parameters are always returned).

➢ Parameter 1:
- ✓ This is the number of measured pulses during the logging interval.
  Format: hexadecimal

➢ Parameter 2:
- ✓ This is the number of measured pulses.
  Format: exponential.

Example of message for Pulse Count Sensors report

➢ Fields:
- ✓ Report ID: This field is set with character 'P' (ECP_PARAM_REPORT_ID).
- ✓ Sensor Type ID: This field is set with character 'P' (ECP_PULSE_CNT_ID).
- ✓ Timestamp: February 25, 2019, 17 hours 30 minutes and 00 seconds is encoded as "19:02:25:17:30:00"
- ✓ SensorID: first pulse sensor attached to TBSL1, its virtual ID is set to character '0'.
- ✓ SubSensorID: pulse input used for rain gauge application, '0'.
- ✓ Number of parameters: 2
- ✓ Parameter 1: number of pulse 0x20 = 32 pulses during the logging period
- ✓ Parameter 2: amount of rain is 6.4mm

➢ So the message for pulse sensor report is as below:

PP19:02:25:17:30:00002 00000020 **6.400000E+00**

- Analog sensor message

Message for Analog Sensors reporting											
Report ID	Sensor Type ID	Timestamp	Sensor ID	SubSensor ID	Nb of parameters	Space	Min Parameter	Space	Avg Parameter	Space	Max Parameter
1 byte	*1 byte*	*17 bytes*	*1 byte*	*1 byte*	*1 byte*	*1 byte*	*5 bytes*	*1 byte*	*5 bytes*	*1 byte*	*5 bytes*

Message's fields:

➢ Report ID
- ✓ Value of this field: character 'P' that is first character of the parameter report message.
- ✓ Field size: 1 byte

➢ Sensor Type ID

- ✓ Format: cf table "Definition for Sensor Identification". Sensor's Type ID for this message is character 'A'.
- ✓ Field size: 1 byte
- ➢ Timestamp
  - ✓ Format: YY:MM:DD:hh:mm:ss (24 hours format)
    - YY: 2 characters representing 2 digits year's number.
    - MM: 2 characters representing 2 digits month's number.
    - DD: 2 characters representing 2 digits date's number.
    - hh: 2 characters representing 2 digits hours number.
    - mm: 2 characters representing 2 digits minutes number.
    - ss: 2 characters representing 2 digits seconds number.
  - ✓ For example: the time stamp October 27, 2016, 13 hours 45 minutes and 00 seconds is encoded as "16:10:27:13:45:00"
- ➢ SensorID
  - ✓ This field is the virtual ID of the analog sensor attached to TBSL1. Value range can be '0', '1', '2', '3', '4', '5', '6', '7', '8' and '9' (Note: on TBSL1 HW revF there are only 2 analog inputs hence SensorID holds either '0' or '1').
  - ✓ Format: decimal
  - ✓ For example:
    - 2nd analog sensor connected to TBSL1, its index is 1
- ➢ SubSensorID
  - ✓ This field is set to '0'.
  - ✓ Format: decimal
- ➢ Number of parameters:
  - ✓ This field is set to '3' as minimum, average and maximum values are returned within this message.
  - ✓ Format: decimal
- ➢ Min Parameter: this is minimum measured value of an analog sensor during measurement interval.
  - ✓ Parameter Format of the Analog Sensor
    Value Format of analog sensor: d.d
    - d – numeric digits before the decimal point
    - . – the decimal point (optional)
    - d – numeric digits after the decimal point

    The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1.
- ➢ Avg Parameter: this is average of measured values of analog sensor during measurement interval.
  - ✓ Parameter Format of the Analog Sensor
    Value Format of analog sensor: d.d
    - d – numeric digits before the decimal point
    - . – the decimal point (optional)
    - d – numeric digits after the decimal point

    The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1.
- ➢ Max Parameter: this is maximum measured value of analog sensor durin measurement interval.
  - ✓ Parameter Format of the Analog Sensor
    Value Format of analog sensor: d.d
    - d – numeric digits before the decimal point
    - . – the decimal point (optional)
    - d – numeric digits after the decimal point

    The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

Example of message for Analog Sensors report

- Fields:
  - ✓ Report ID: This field is set with character 'P' (ECP_PARAM_REPORT_ID).
  - ✓ Sensor Type ID: This field is set with character 'A' (ECP_ANALOG_ID).
  - ✓ Timestamp: February 26, 2019, 18 hours 15 minutes and 00 seconds is encoded as "19:02:26:18:15:00"
  - ✓ SensorID: Index of 2nd analog sensor is as character '1'.
  - ✓ SubSensorID: set to '0'
  - ✓ Number of parameters: '3'
  - ✓ Min Parameter: 1.249
  - ✓ Avg Parameter: 1.250
  - ✓ Max Parameter: 1.251
- So the analog sensor reporting message is as below:
  ***PA19:02:26:18:15:00103* 1.249 1.250 1.251**

- SDI-12 sensor message

Parameter Report Message of SDI-12 Sensor										
Report ID	Sensor Type ID	Time stamp	SDI-12 Probe ID	SDI-12 Ordinal	Number of Parameters	Space Character	Parameter 1	….	Space Character	Parameter n
1 byte	*1 byte*	*17 bytes*	*1 byte*	*1 byte*	*2 bytes*	*1 byte*	*variable*	…	*1 byte*	*variable*

Message's fields:

- Report ID
  - ✓ Value of this field: character 'P' that is first character of the parameter report message.
  - ✓ Field size: 1 byte
- Sensor Type ID
  - ✓ Format: cf table "Definition for Sensor Identification". Sensor's Type ID for this message is character 'S'.
  - ✓ Field size: 1 byte
- Timestamp
  - ✓ Format: YY:MM:DD:hh:mm:ss (24 hours format)
    - YY: 2 characters representing 2 digits year's number.
    - MM: 2 characters representing 2 digits month's number.
    - DD: 2 characters representing 2 digits date's number.
    - hh: 2 characters representing 2 digits hours number.
    - mm: 2 characters representing 2 digits minutes number.
    - ss: 2 characters representing 2 digits seconds number.
  - ✓ For example: the time stamp October 27, 2016, 13 hours 45 minutes and 00 seconds is encoded as "16:10:27:13:45:00"
- SDI-12 Probe ID:
  - ✓ This is the virtual ID of an SDI-12 sensor which can be composed of several SDI-12 sub-sensors (eg: a SDI-12 soil moisture probe made of 10 cells, a SDI-12 weather station integrating SDI-12 wind speed sensor and solar radiation sensor, etc…).
  - ✓ Value format: there is one character with range from '0' to '9'.
- SDI-12 Ordinal
  - ✓ The sensor ordinal is the index of a SDI-12 command of corresponding SDI-12 sub-sensor in a probe (eg the 3rd SDI-12 soil moisture cell in a soil moisture probe).
  - ✓ Format: this is one character which ranges from '0' to '9' and from 'a' to 'z'.
  - ✓ Examples:

The SDI-12 ordinal is number 3 that will be encoded as ‘2’.

The SDI-12 ordinal is number 13 that will be encoded as ‘c’.

- ➢ Number of Parameters
  - ✓ These are two characters which represent the number of measurements returned by the SDI-12 command, and each characters is one of ‘0’, ‘1’, ‘2’, ‘3’, ‘4’, ‘5’, ‘6’, ‘7’, ‘8’ and ‘9’.
  - ✓ Size of this field: 2 bytes
  - ✓ For example: SDI-12 sub-sensor returns 6 values, the number of parameters is encoded as ‘06’.
- ➢ Space Character: used as a delimiter.
- ➢ Parameter x

  Format: pd.d
  - ✓ p – the polarity sign (+ or -)
  - ✓ d – numeric digits before the decimal place
  - ✓ . – the decimal point (optional)
  - ✓ d – numeric digits after the decimal point

  The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1. The maximum number of characters in a data value is 9 (the (polarity sign + 7 digits + the decimal point)).

Example of message for SDI-12 Sensor report

- ➢ Fields:
  - ✓ Report ID: This field is set with character ‘P’ (ECP_PARAM_REPORT_ID).
  - ✓ Sensor Type ID: this field is set with character ‘S’ (ECP_SDI12_ID).
  - ✓ SDI-12 Probe ID: this is the virtual ID of the, set to ‘0’.
  - ✓ SDI-12 Ordinal: ordinal value of sub-sensor is set to ‘3’.
  - ✓ Timestamp: October 29, 2016, 14 hours 30 minutes and 00 seconds is encoded as “16:10:29:14:30:00”.
  - ✓ Number of parameters: SDI-12 sub-sensor returns 4 measurements, and this is encoded as character ‘04’.
  - ✓ Parameter x: there are values: -12.20003, 2.322432, -4.433332 and -9.110423 that are encoded as string: “-12.20003”, “+2.322432”, “-4.433332” and “-9.110423”.
- ➢ So the message for SDI-12 sensor report is as below:

  PS16:10:29:14:30:000304 -12.20003 +2.322432 -4.433332 -9.110423

- Battery level message

Message Format of Battery Parameter Report				
Report ID	Sensor Type ID	Timestamp	Space Character	Battery Voltage
1 byte	*1 byte*	*17 bytes*	*1 byte*	*5 bytes*

Message’s Fields:

- ➢ Report ID
  - ✓ Value of this field: character ‘P’ that is the first character of the parameter report message.
  - ✓ Field size: 1 byte
- ➢ Sensor Type ID
  - ✓ Format: cf table “Table 8 3: Sensor Identification in the Protocol”. Sensor’s Type ID for this message is character ‘B’.

- ✓ Field size: 1 byte
- ➢ Timestamp
  - ✓ Format: YY:MM:DD:hh:mm:ss (24h format)
    - o YY: 2 characters representing 2 digits year's number.
    - o MM: 2 characters representing 2 digits month's number.
    - o DD: 2 characters representing 2 digits date's number.
    - o hh: 2 characters representing 2 digits hours number.
    - o mm: 2 characters representing 2 digits minutes number.
    - o ss: 2 characters representing 2 digits seconds number.
  - ✓ Field size: 17 bytes
  - ✓ For example: the time stamp October 27, 2016, 13 hours 45 minutes and 00 seconds is encoded as "16:10:27:13:45:00".
- ➢ Battery Voltage
  - ✓ Format of battery voltage: d.d
    - o d – integer part (numeric digits)
    - o . – the decimal point (optional)
    - o d – decimal part (numeric digits)

    The maximum number of digits for a data value is 3, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1.
  - ✓ Field size: 5 bytes
  - ✓ Example: value of the battery voltage is 2.36 which will be encoded as "2.360".

Example of message for battery parameter report

- ➢ Fields:
  - ✓ Report ID: This field is set with character 'P' (ECP_PARAM_REPORT_ID).
  - ✓ Sensor Type ID: This field is set with character 'B' (ECP_BAT_VOL_ID).
  - ✓ Timestamp: October 27, 2016, 14 hours 45 minutes and 00 seconds is encoded as "16:10:27:14:45:00".
  - ✓ Parameter: battery voltage 4.1 V, encoded as '4.100'.
- ➢ So the message for pulse sensor report is same as this string: ***PB16:10:27:14:45:00*** **4.100**

3.2.4 Uplink data format (TBSL1/RFB-4GWW, TBSL1/RFB-NbIOT-CatM1, TBSL1/RFB-Wifi)

Packets are published as a JSON object. The topic is configurable through the PC configuration tool.

Here below is an example of JSON payload:


```
{"msgNumber":3,"payload":"PA18:11:14:15:40:001 0.713 0.721 0.723","modemType":"Cellular","eui":"00000101"}
```


JSON payload format:

{

"msgNumber" : <Decimal>,

"payload" : <ASCII payload>,

"modemType":"Cellular",

"eui" : <Hexadecimal EUI>

}

- <msgNumber>: ID of the published message. Starts from zero and is increased upon each published message. This parameter is used to identify duplicated messages.
- <payload>: contains the measurements or reporting payload. Format is the same than for LoRaWAN uplink packets.
- <modemType>: set to “Cellular” – may have other values in the future depending on the supported modem.
- <eui>: this is the EUI/ID of the platform, hexadecimal string.

3.2.5 Uplink Javascript payload parser implementation example

A portable way to parse uplink payload which is almost not language related is to use regular expressions (regex).

The below sample code is written in Javascript but could be easily adapted for other languages (a good regex online tester and verifier is [Regex101]).

The subsequent sections lists all Javascript regular expressions needed to extract required fields from the payload. They are provided for information only and shall be modified based on user’s needs.

3.2.5.1 *Sample code*

The following sample code parses *payload* string and extracts the message type into *msgtype* capturing group. The result is displayed to the console.

```
// SDI-12 payload
const payload = 'PS20:03:12:02:45:000208 +0.317 +0.64 +0.559 +0.459 +0.666 +0.881 +0.611 +0.654';

// Regular expression to extract the message type
const regex_filter = /^(?<msgtype>PS|PB|PA|PP|C).*/;

// Execute the regular expression over the payload string
const result = regex_filter.exec(payload);

// Display the capturing group msgtype to the console
console.log(result.groups.msgtype);
```

This code can be easily customized depending on the payload type by using following regular expressions to extract required fields.

3.2.5.1 *Message type*

Get message type: PS/PB/PA/PP/C:

/^(?<msgtype>PS|PB|PA|PP|C).*/

3.2.5.2 *Time stamp*

Extract the year, month, day, hour and minutes from the time stamp:

```
/^(?:\D\D{0,1})(?<year>\d\d)(?:\:)(?<month>\d\d)(?:\:)(?<day>\d\d)(?:\:)(?<hour>\d\d)(?:\:)(?<min>\d\d)(?:.*)/
```

3.2.5.3 *Sensor ID*

Get the sensor index, only applicable to PS/PA/PP message types.

```
/^(?:PS|PA|PP)(?:.{17})(?<sensor>\d)(?:.*)/
```

3.2.5.4 *Sub-Sensor ID*

Get the sub-sensor index, only applicable to PS/PA/PP message types.

```
/^(?:PS|PA|PP)(?:.{18})(?<subsensor>\d)(?:.*)/
```

3.2.5.5 *Battery level*

Battery level is extracted from PB message.

```
/^(?:PB)(?:.{18})(?<battery>.*)/
```

3.2.5.6 *RSSI*

RSSI is extracted from C message.

```
/^(?:C)(?:.{37})(?<rssi>.*)/
```

3.2.5.7 *Number of measurements (PS)*

Get number of measurements for a PS message.

```
/^(?:PS)(?:.{19})(?<number>\d\d)(?:.*)/
```

3.2.5.8 *Measurements (PS)*

Get all measurements held by a PS message.

```
/^(?:PS)(?:.{22})(?<payload_temp>.*)/
```

Regex expressions to get the first 10 measurements one by one (*val1* to *val10*):

```
/^(?:PS)(?:.{22})(?<val1>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){1})(?<val2>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){2})(?<val3>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){3})(?<val4>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){4})(?<val5>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){5})(?<val6>[^\s]{1,9})(?:.*)/
```

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){6})(?<val7>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){7})(?<val8>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){8})(?<val9>[^\s]{1,9})(?:.*)/

/^(?:PS)(?:.{22})(?:([^\s]+[\s]){9})(?<val10>[^\s]{1,9})(?:.*)/

3.2.5.9 *Number of measurements (PP)*

Get number of measurements for a PP message.

/^(?:PP)(?:.{19})(?<number>\d)(?:.*)/

3.2.5.10 *Measurements (PP)*

Get all measurements held by a PP message.

/^(?:PP)(?:.{21})(?<payload_temp>.*)/

Regex expressions to get the 2 measurements of a PP message:

/^(?:PP)(?:.{21})(?<pulse>[^\s]{1,8})(?:.*)/

/^(?:PP)(?:.{21})(?:([^\s]+[\s]){1})(?<totalizer>[^\s]{1,12})(?:.*)/

Note: pulse is a hexadecimal value and the totalizer is a float.

3.2.5.11 *Number of measurements (PA)*

Get number of measurements for a PA message.

/^(?:PA)(?:.{19})(?<number>\d)(?:.*)/

3.2.5.12 *Measurements (PA)*

Get all measurements held by a PA message.

/^(?:PA)(?:.{21})(?<payload_temp>.*)/

Regex expressions to get the 3 measurements of a PA message:

/^(?:PA)(?:.{21})(?<val1>[^\s]{1,8})(?:.*)/

/^(?:PA)(?:.{21})(?:([^\s]+[\s]){1})(?<val2>[^\s]{1,12})(?:.*)/

/^(?:PA)(?:.{21})(?:([^\s]+[\s]){1})(?<val3>[^\s]{1,12})(?:.*)/

3.2.6 Downlink data format (TBSL1/RFB-LoRa)

TBSL1 provides remote control of the platform through downlink messages.

The downlink control API allows following control on TBSL1/RFB-LoRa platform:

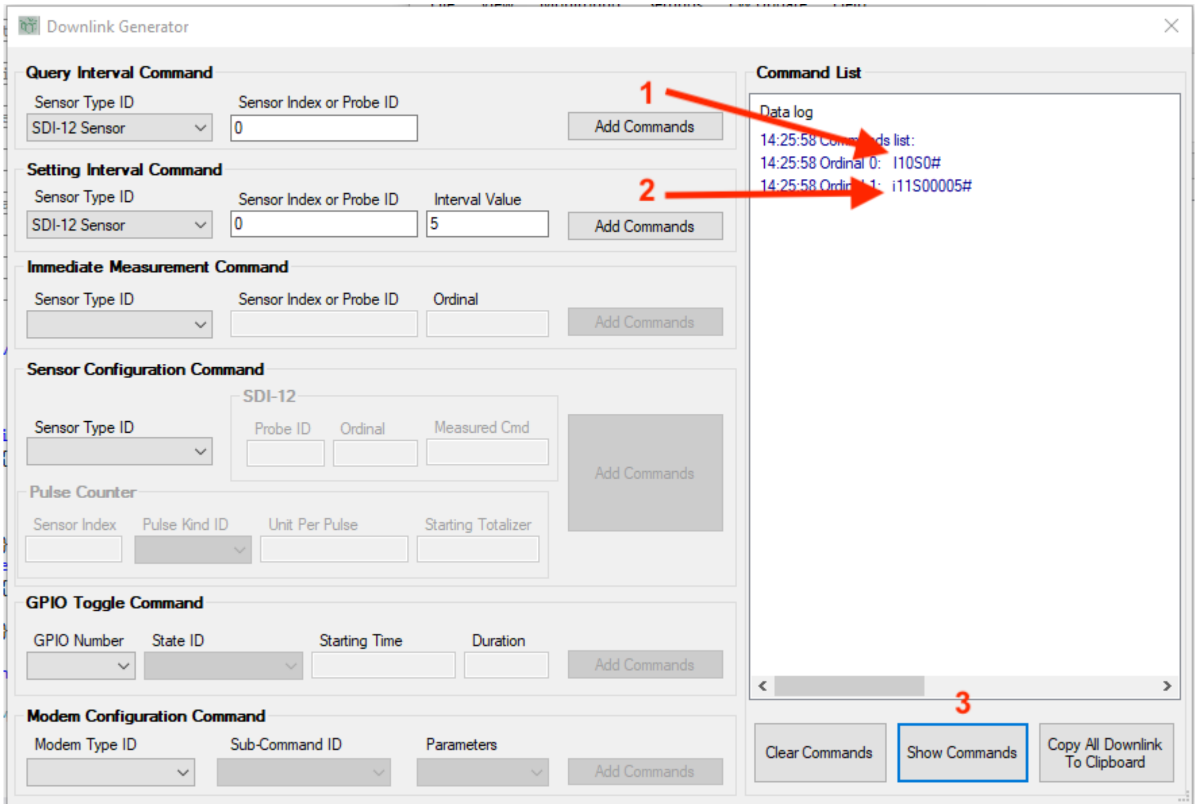
- Query and update measurement intervals
- Query and update transmission interval
- Update SDI-12 sensors commands
- Perform immediate sensor measurement
- Trigger GPIO to act as a switch to control relays (*provisioned for a later HW upgrade*)
- Configure internal alarms
- Configure pulse sensor parameters
- Change LoRaWAN Class (A / C)
- Enable or disable LoRaWAN ADR
- Update TBSL1 RTC time

The remote server needs to send downlink control packet to the LoRaWAN server.

The PC configuration tool includes a feature that allows to automatically generate the right downlink commands to ease the integration at application server side (refer to chapter *4.5 Downlink Generator* from the TBSL1 PC configuration tool user guide).

Commands must be grouped before being sent by the remote server to TBSL1. Each group can contain up to 16 commands, indexed from ‘0’ to ‘F’. It is therefore crucial to generate the final list of commands by clicking on the ‘Show Commands’ button from the configuration tool:

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com



- First define the commands that need to be generated and click on ‘Add Commands’ for each of them (step 1&2 in above example)
- Then click on ‘Show commands’ (step 3) to generate the downlink commands to be sent
  - *Note: commands for both LoRaWAN and cellular (JSON format) are displayed; copy/paste to the clipboard and keep relevant command list depending on the modem that is used.*
- Send each command on the LoRaWAN downlink channel from the LoRaWAN server.

Before proceeding with another group of commands the remote server needs to wait that TBSL1 has fully processed the previous commands batch.

For this it has to monitor the acknowledgement messages returned by TBSL1 for each command. It’s up to the remote server to handle the retransmission of message whose acknowledgement contains an error information.

3.2.5.13 *Downlink command request format (automatically generated with TBSL1 downlink generator tool)*

<Command_ID><Max index in the commands group><Command index><parameters><!> where:

- Command_ID
- Max index in the commands group: ‘0’ to ‘F’ => ‘0’ means 1 command in the group, ‘F’ means 16 commands in the group.
- Command index: ‘0’ to ‘F’, ie 0 to 15.
- Parameters: dependant on the command type.
- ‘!’: End response termination.

3.2.5.14 *Downlink command response format*

The generic format is the following:

Response Format				
Command ID	CMD Ordinal	Error Code	Data	End Response Character
1 character	*1 character*	*1 character*	*Variable*	*1 character*

<Command_ID> list:

Values *(characters)*	Type	Description
'M'	Measurement	Request a sensor to execute an immediate measurement.
'I'	Query	Query measurement or transmission interval value.
'i'	Configuration	Change a sensor's measurement interval time or report transmission interval.
'm'	Configuration	Change measurement and data commands of SDI-12 sensors.
'a'	Configuration	Change measurement configuration of analog sensors.
'p'	Configuration	Change measurement configuration of pulse sensors.
'o'	Configuration	Change modem configuration.
'g'	Setting	Toggle GPIO on the platform.

Note: this parameter is set to 'X' in case of error code 6 (lost command, cf below).

<*CMD Ordinal*>: command's index in the group.

<error code>:

Values *(characters)*	Description
'0'	Valid command has been received and successfully processed.
'1'	Received command has an invalid Command ID.
'2'	Received command has missing or redundant parameters.
'3'	Received command has format errors.
'4'	Command message's length is too long.
'5'	Duplicated received command.
'6'	Lost command.
'7'	The end device has EEPROM corruption after max number of retries (during erase, program, read process, etc.)

<data>:

SDI-12 immediate measurement command response: data format

Data Format of SDI-12 Immediate Measurement Command				
Timestamp	Parameters Number (n)	Parameter 1	…	Parameter n
12 characters	*1 character*	*variable*	*variable*	*variable*

Message's fields:

- Timestamp
  - ✓ This is the time at which the node received the command and executed the requested SDI-12 measurement.
  - ✓ Format: timestamp is encoded as "YYMMDDhhmmss".
  - ✓ Field Size: 12 bytes
- Parameters Number (n)
  - ✓ This is the number of parameters that will be returned to application server. The number is encoded as a character from '1' to '9'.
  - ✓ Field Size: 1 byte
- Parameter 1
  - ✓ This is the value of SDI-12 sensor's measurement. The value's format is as below:
    Format: - pd.d
    - p – the polarity sign (+ or -)
    - d – integer part (numeric digits)
    - . – the decimal point (optional)
    - d – decimal part (numeric digits).
    - The maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 1. The maximum number of characters in a data value is 9 (the (polarity sign + 7 digits + the decimal point)).

    Field Size: variable

Example: send command to perform an immediate measurement with probe ID '0', ordinal 2 and timestamp as 18:01:09 15:25:00 and command session (command ordinal: 5, number of commands: 15). So the command string will be sent as below:

- Command: "Mf5S02#"
- Response: "M501801091525002+1.3121213-12.8674863!"

Pulse immediate measurement command response: data format

Data Format for Pulse Sensor	
Timestamp	Parameter
12 characters	*variable*

Message's Fields:

- Timestamp
  - ✓ This is the time at which the node received the command and executed the requested SDI-12 measurement.
  - ✓ Format: timestamp is encoded as "YYMMDDhhmmss".
- Parameter: d.d
  - ✓ d – integer part (numeric digits)
  - ✓ . – the decimal point (optional)
  - ✓ d – decimal part (numeric digits)
  - ✓ The number of digits ranges from 1 to 7 (excluding decimal point).
  - ✓ Field size: variable

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

Example: send command to perform an immediate pulse measurement with index 0; response message with measurement value 544345.3, timestamp 18:01:09 15:25:00 and command session (command ordinal: 10, number of commands: 15). Command and response messages are then as below:

- Command: “MfaP0#”
- Response: “Ma0180109152500544345.3!”

Analog immediate measurement command response: data format

Data Format for Analog Sensor			
Timestamp	Min Value	Avg Parameter	Max Value
12 characters	*variable*	*variable*	*variable*

Message’s fields:

- Timestamp
  - ✓ This is the time at which the node received the command and executed the requested SDI-12 measurement.
  - ✓ Format: timestamp is encoded as “YYMMDDhhmmss”.
- Parameter: pd.d
  - ✓ p – polarity sign ‘+’.
  - ✓ d – integer part (numeric digits)
  - ✓ . – the decimal point (optional)
  - ✓ d – decimal part (numeric digits)
  - ✓ maximum number of digits for a data value is 7, even without a decimal point. The minimum number of digits for a data value (excluding the decimal point) is 7. The maximum number of characters in a data value is 9 (the (polarity sign + 7 digits + the decimal point)).

  Field Size: variable

Example: send command to immediate measure parameter of analog with index 1, parameters ((min, avg, max) = (1.312121, 1.512121, 2.312121)), timestamp as 18:01:09 15:25:00 and command session (command ordinal: 11, number of commands: 15). So the command string will be to send as below:

- Command: “MfbA1#”

Response: “Mb0180109152500+1.312121+1.512121+2.312121!”

Setting interval command response: data format

For this command data format field is not relevant.

Query interval command response: data format

- This is sensor’s time interval
  - ✓ Value of this field is number of minutes represented over 4 characters.
  - ✓ Size of this field: 4 bytes.
  - ✓ Example: an interval of 1380 (hexadecimal format: 0x0564) minutes is represented as characters string “0564” in the command format.

SDI-12/Pulse/Analog/Modem/GPIO configuration command response: data format

For this command data format field is not relevant.

End response character: '!'

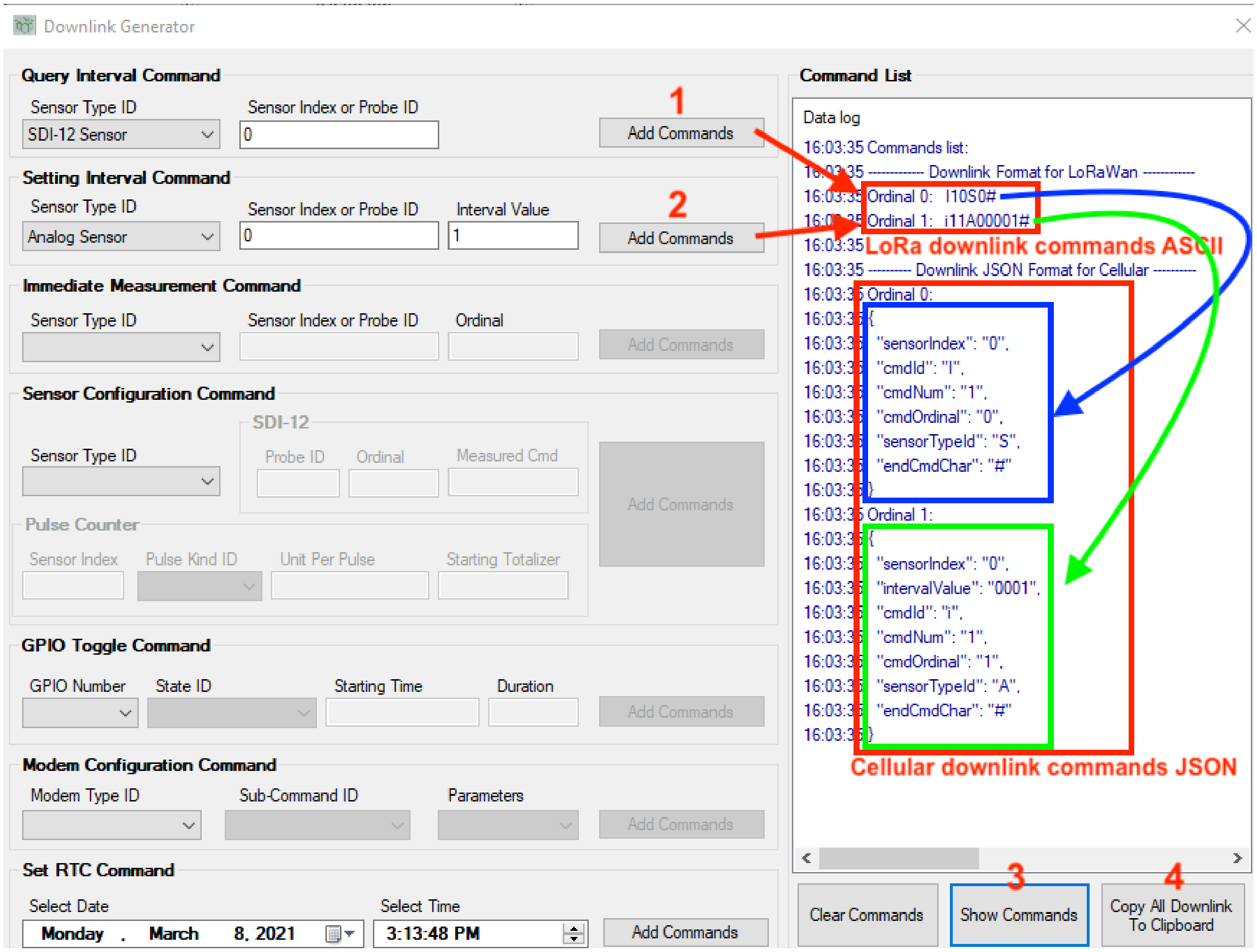
3.2.7 Downlink data format (TBSL1/RFB-4GWW, TBSL1/RFB-NbIOT-CatM1, TBSL1/RFB-Wifi-DC)

TBSL1 provides remote control of the platform through downlink messages.

The downlink control API allows following control on TBSL1/RFB-4GWW platform:

- Query and update measurement intervals
- Query and update transmission interval
- Update SDI-12 sensors commands
- Perform immediate sensor measurement
- Trigger GPIO to act as a switch to control relays (*provisioned for a later HW upgrade*)
- Configure internal alarms
- Configure pulse sensor parameters

The remote server needs to publish control messages to the downlink topic that has been configured with the PC configuration tool and using the JSON command format:



- First define the commands that need to be generated and click on ‘Add Commands’ for each of them (step 1&2 in above example)
- Then click on ‘Show commands’ (step 3) to generate the JSON objects corresponding to each command
- Copy to clipboard (step 4) to keep the desired command list (LoRa or Cellular)
- Publish each JSON object to the configured TBSL1 Downlink Topic.

The PC configuration tool includes a feature that allows to automatically generate the right downlink commands to ease the integration at application server side.

The downlink command responses are the same than for LoRaWAN but are returned within a JSON structure:

{
“msgNumber” : ***0****,*
*“payload” : “****Downlink_command_response****”,*
“modemType” : “Cellular”,
*“eui” : “****00000101****”*
}

With:

- *msgNumber*: index of the command which is responded to.
- *payload*: the command's response
- *modemType*: currently always set to 'Cellular'
- *eui*: TBSL1 EUI (8 characters)

3.2.8 Alarm programming

TBSL1 integrates an embedded alarm feature that is configured through PC application.

It allows programming up to 2 different alarms on any attached sensor (SDI-12, analog or pulse) that is triggered internally in the platform whenever one of the programmed condition is met:

- Below threshold
- Above threshold
- Inside limits
- Outside limits

The user can define on which value measured by a given sensor the alarm should be triggered (eg for a sensor that returns both air humidity and temperature, alarm can be configured to be generated either on humidity or temperature measurement).

When alarm is triggered, one of the programmed action is executed:

- Warning message can be sent to the server
    - As a packet with TBSL1/RFB-LoRa
    - As a message published over MQTT/HTTP/FTP with TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1

3.2.9 Rain gauge and flow meter applications

The pulse input can be used for either rain gauge or flow meter applications.

TBSL1 provides up to 2 pulse inputs, each of them is configured through the PC application to operate as rain gauge or flow meter.

- Rain gauge mode
    - The capacity per cup (in mm) is configured through PC application
    - TBSL1 reports on each period the number of pulses and the number of pulses multiplied by the capacity.
- Flow meter mode
    - The dials on the front of the water meter is used to initialize the Flow Totalizer.
    - Capacity is indicated in liter
    - TBSL1 reports the number of pulses and the totalizer reading at the end of the logging period (ie it keeps on increasing).

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

4 TBSL1 configuration

4.1 Overview

Before being used, TBSL1 must be configured using the TBSL1 PC configuration tool:

- Radio and network parameters:
  - TBSL1/RFB-LoRa:
    - LoRaWAN ciphering keys and identifiers
    - Enrollment mode (ABP or OTAA)
    - LoRa frequencies and data rates
    - LoRaWAN REPEAT and ACK features
    - Etc…
  - TBSL1/RFB-4GWW, TBSL1/RFB-NbIOT-CatM1:
    - RAT selection
    - Export method (MQTT, HTTP, FTP) and related parameters
    - Security parameters (certificates, user name/password)
  - **Note for TBSL1/RFB-4GWW and TBSL1/RFB-NbIOT-CatM1: SIM card must be inserted and antenna connected before configuring the platform.**
  - TBSL1/RFB-Wifi:
    - Wi-Fi network selection
    - Connection credentials
    - MQTT parameters configuration
- Sensors configuration:
  - Defines the type and number of sensors that are connected to the platform
  - Defines sensors specific parameters (eg SDI-12 commands to be used, rain gauge capacity for volume computation, temperature alarm threshold, etc…)
  - 3 kind of sensors are supported by the platform:
    - SDI-12
    - Analog
    - Pulse
- Operation intervals:
  - Measurement intervals for each sensor
  - Wireless transmission interval
  - Reporting/Monitoring interval
  - Program actual real time into the platform (UTC aligned or based on local time zone)
- …

TBSL1 configuration is achieved through a PC application that accesses the platform through its external USB port, the platform is in console mode.

To switch TBSL1 to logging mode (i.e. in the mode where it will take measurements, sleep and transmit), it is required to either select the "logging mode" option from the configuration tool or make the platform hibernate: upon reset it will automatically start in logging mode.

Refer to the TBSL1 PC configuration tool user guide for further information regarding the configuration and the modes.

4.2 Limitations

The system has following limitations in terms of sensors configuration:

- Analog sensors
  - 2 channels maximum
- Pulse sensors:
  - 1 channel maximum
- SDI-12 sensors:
  - 10 probes (different SDI-12 addresses can be used for a given probe however a given SDI-12 address can't be used on different probes)
  - Maximum 40 SDI-12 measurement commands in total for all probes
  - Refer to TBSL1 configuration tool user manual for further information.

4.3 Time synchronization

TBSL1 embeds a precision RTC element that is backed up by a Gold capacitor to retain the real time while TBSL1 is powered off. Should TBSL1 be left on shelves unpowered for more than 3 months, it is strongly recommended to power it before that limit is reached. Letting it powered for few minutes allows to recharge the Gold capacitor and to avoid loosing the real time settings.

However due to crystal tolerance and temperature, it may happen TBSL1 time drifts over time.

It is possible to enable an option in the configuration tool that allows resynching the internal clock when the daily drift time is above a specified value.

The resynchronization procedure differs depending on the installed modem:

- LoRaWAN
  - *DeviceTimeReq* MAC command is sent to the LoRaWAN server to request the actual time.
- Cellular
  - The cellular modem automatically updates TBSL1 internal time with the time provided by the cellular network.
- Wi-Fi
  - TBSL1 gets the actual time from a NTP server (pool.ntp.org)

5 Power Management

5.1 Power saving modes

The platform is programmed so each sensor takes measurements on its own timing interval, and measurement data transmission happens on a configurable interval. Outside these periods of measurement and transmission activities, the platform enters in sleep mode lowering the platform's current consumption to its minimum.

While sleeping, the platform can be waken up when one of these events occur:

- Sensor measurement event
- Transmission event
- External pulse sensor event

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

Note: when USB is connected to the platform, the system doesn't enter into sleep mode.

5.2 Battery charging protection

These restrictions apply only to TBSL1 solar version.

TBSL1 implements a battery charging protection mechanism.

While TBSL1 can operate over -40 to +85 degrees, regular Li-Ion batteries won't work at negative temperatures and may explode at high temperatures. This is why TBSL1 charging protection circuit will only charge the battery when the temperature is between zero and 50 degrees Celcius.

In most cases, this is not an issue as TBSL1 will be still be powered through both battery and solar panel during day time and the user should not be concerned about it.

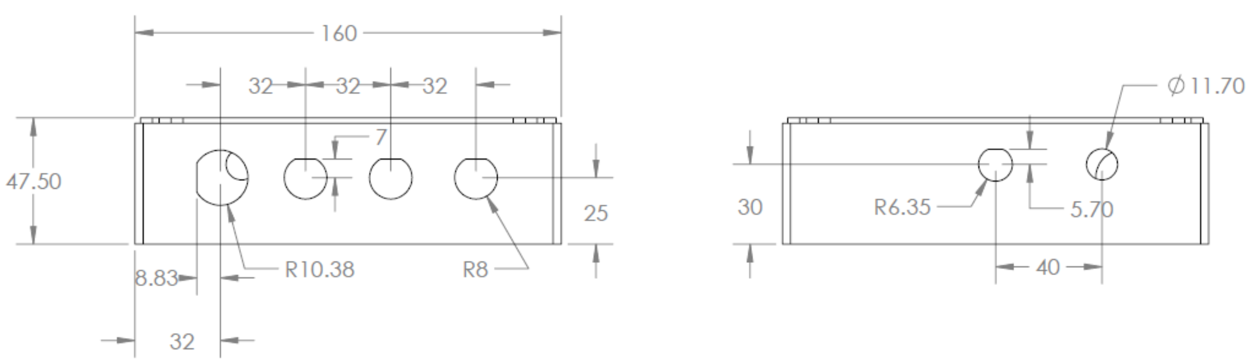
However in case TBSL1 would have to be used over several weeks outside of 0/+50 degrees range, then it could lead to the battery draining out depending on the periodicity of sensors measurements and transmission.

Would there be a specific need for TBSL1 to allow battery charging outside 0/+50 degrees range, please contact Tekbox sales (sales@tekbox.com) for further information.

6 Mechanical Specifications and mounting instructions

TBSL1 is suitable for outdoor usage thanks to its IP67 aluminium housing: SP-AG-FA3-1 family, single piece gasket, IP67, TNC antenna connector, Gore-Tex vent, IP67 circular panel connectors.

TBSL1 dimensions:



Connectors references to connect sensors and solar panel to the front panel:

- Solar panel: M14 5 pins female assembly connector, LLT-M14CM05
- Sensors: M14 7 pins female assembly connector, LLT-M14CM07

Solar panel connectivity:

BATIN and BATOUT pins of solar panel connector must be shorted to allow battery power saving.

When the solar panel is not plugged the platform is not powered as the battery is not connected internally to the power circuit.

When the solar panel is connected and pins 3&4 are shorted, the battery is automatically connected to the power circuit and the unit starts running on battery and solar panel power supplies.

If the unit has previously been switched to logging mode, it will then start its measurements/sleep/transmission sequences.

Note: these pins are already shorted in the solar panels purchased from Tekbox.

BATTERY SWITCH

7 SDI-12 Basics

SDI-12 is a serial data communication standard for interfacing multiple sensors with a data recorder The SDI-12 uses a shared bus with 3 wires: power (12V), data, ground
Data rate: 1200 baud

Each sensor at the bus gets a unique address which is in the range ASCII [0-9, a-z, A-Z]. The default address of every sensor is ASCII[0]. When setting up a SDI-12 sensor network, every sensor needs to be configured with a unique address. This can be done using the *"Change Address Command"*.

A sensor typically can measure one or more parameters. Sensor manufacturers usually specify *"Extended Commands"* to configure or calibrate sensors. These commands are specified by the manufacturer, but they follow the command structure specified by SDI-12.

A typical recorder/sensor measurement sequence proceeds as follows:
1) The data recorder wakes all sensors on the SDI-12 bus with a break.
2) The recorder transmits a command to a specific, addressed sensor, instructing it to make a measurement.
3) The addressed sensor responds within 15.0 milliseconds returning the maximum time until the measurement data will be ready and the number of data values it will return.
4) If the measurement is immediately available, the recorder transmits a command to the sensor instructing it to return the measurement result(s). If the measurement is not ready, the data recorder waits for the sensor to send a request to the recorder, which indicatesthat the data are ready. The recorder then transmits a command to get the data.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

5) The sensor responds, returning one or more measurement results.

SDI-12 command structure:
Each SDI-12 command is an ASCII string with up to 5 characters, starting with the sensor address and terminated by a **!** character.

Example:
Send Identification Command **0I!**
0 is the sensor address (sensor zero). Upon receiving this command, the sensor will send an ASCII string containing sensor address, a SDI-12 compatibility number, company name, sensor model number, sensor version number and sensor serial number.

The standard process to carry out a measurement is to send a measurement request upon which the sensor responds with the time that is required to carry out the measurement and the number of data items being returned. After waiting the time that the sensor requires to carry out the measurement, the data recorder sends a *"Read Command"* to get the measurement results.

Example:
Start Measurement Command **0M1!**
Sensor 0 might respond 00302 which means the measurement will take 30 seconds and deliver 2 values.
After min. 30 seconds, the data recorder can send the Read Data Command **0D0!** To which Sensor 0 might reply 0+27+1050. +27+1050 is the two measurement results which may be 27°C and 1050 milibar.
The response string of a sensor is always in ASCII format and may contain up to 40 or up to 80 characters, depending on the type of command. Out of 40 or 80 characters, the values part of the response string may contain up to 35 or 75 characters.
The response string of a sensor is always in ASCII format and may contain up to 40 or up to 80 characters, depending on the type of command. Out of 40 or 80 characters, the values part of the response string may contain up to 35 or 75 characters.

8 LoRaWAN network basics

8.1 Overview

This section provides the TBSL1/RFB-LoRa user a basic understanding of LoRaWAN key features, so TBSL1/RFB-LoRa can be integrated smoothly in such ecosystem.

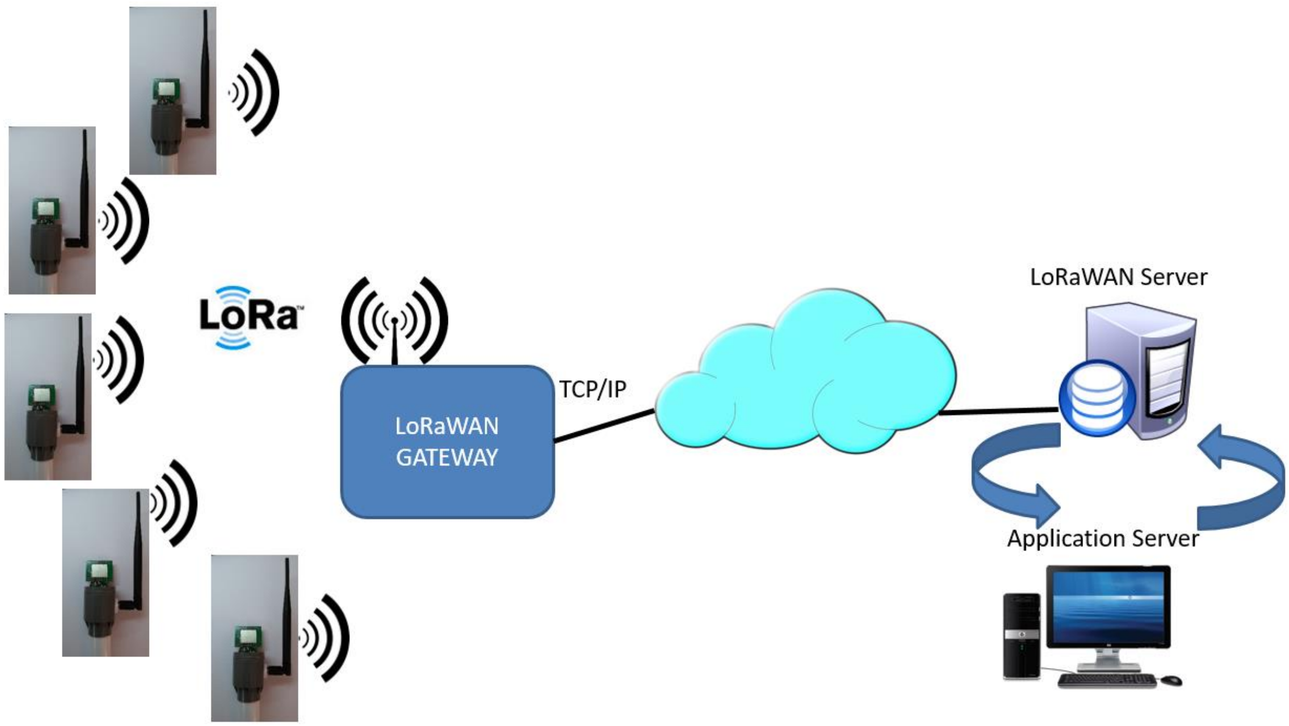
LoRa Alliance describes a LoRaWAN network as follows:
"

LoRaWAN network architecture is typically laid out in a star-of-stars topology in which **gateways** is a transparent bridge relaying messages between **end-devices** and a central **network server** in the backend. Gateways are connected to the network server via standard IP connections while end-devices use single-hop wireless communication to one or many gateways. All end-point communication is generally bi-directional, but also supports operation such as multicast enabling software upgrade over the air or other mass distribution messages to reduce the on air communication time.
Communication between end-devices and gateways is spread out on different **frequency channels** and **data rates**. The selection of the data rate is a trade-off between communication range and message duration. Due to the spread spectrum technology, communications with different data rates do not interfere with each other and create a set of "virtual" channels increasing the capacity of the gateway. LoRaWAN data rates range from 0.3 kbps to 50 kbps. To maximize both battery life of the end-devices and overall network capacity, the

LoRaWAN network server is managing the data rate and RF output for each end-device individually by means of an **adaptive data rate** (ADR) scheme.

“

This leads then to a network where each TBSL1/RFB-LoRa is a unique end node communicating with a gateway as highlighted in below schematics:



Several components are involved in the LoRaWAN network:

- End nodes: this is any end-user device integrating a LoRaWAN modem and communicating with a LoRaWAN gateway. With respect to this, TBSL1/RFB-LoRa is a end node.
- Gateway: it acts as a packet forwarder between end nodes and LoRaWAN server. It communicates through LoRa radio with end nodes, and through a TCP/IP connection with LoRaWAN server (depending on the gateway capability, this can be achieved with an Ethernet, wifi or cellular connection). One gateway can accommodate thousands of end nodes, and typical range is 2km in urban areas and around 15km in rural areas with clear line of sight.
- LoRaWAN server: this is a service provided by a 3rd party company (hence the user needs to subscribe to such service). As LoRaWAN packets are encrypted, the LoRaWAN server proceeds with deciphering of LoRaWAN packets and make them available to user’s application server through various communication protocols (this is totally dependant on the LoRaWAN service provider, but HTTPS, WebSocket and REST are widely used protocols besides direct integration with IoT platforms like Microsoft Azure or AWS IoT). An API is then provided to access deciphered data, device EUI and other radio parameters (usually in JSON or XML format).

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

8.2 Integrating TBSL1/RFB-LoRa in a LoRaWAN ecosystem

To build a private network of TBSL1/RFB-LoRa RTUs, it is first required to choose a LoRaWAN gateway (eg www.kerlink.com, www.multitech.com, …) and a LoRaWAN services provider (eg www.loriot.io, www.thethingsnetwork.org…).

TBSL1/RFB-LoRa is compatible with any LoRaWAN certified gateway and network service provider.

This choice is dependant on several criteria:

- Geographic area where the sensors will be deployed: this determines the LoRaWAN frequency plan to be used. The gateway must then have required HW to support these frequencies (eg 868MHz, 915MHz, etc…)
- Gateway connectivity: depending on where the gateway will be installed, next options need to be considered,
  - Indoor / outdoor model
  - Connectivity: Ethernet, cellular, wifi
- The LoRaWAN service provider:
  - Needs to support the chosen gateway (because it will be programmed to access the selected LoRaWAN service provider)
  - Must support the frequency plans required by the user
  - Must support the LoRaWAN features required by the user (eg LoRaWAN downlink, Class C, OTAA, etc…)
  - Provides a suitable interface for the user so the application server can collect deciphered data (eg JSON/XML API reachable through Websocket, REST, etc…)
  - Must be ideally located in the same region where end nodes are deployed, to avoid latenties (eg end nodes in Vietnam, LoRaWAN server in Singapore but not in Germany)
  - Support of multiple applications/accounts, scalability of the server, …
  - Pricing model: subscription based, billing per gateway and end nodes, etc…

The next step is to activate each end node so they are identified and can communicate with the LoRaWAN network.
LoRaWAN standard defines 2 ways of activating end nodes:

- Activation By Personalization (ABP): with this configuration, the end node is bound to a specific LoRaWAN network. This mode is supported by default in TBSL1/RFB-LoRa, and can be compared to a smartphone that is SIM-locked to a specific cellular network.
- Over the Air Activation (OTAA): It gives the end node the “roaming” capability, ie the end node is not bound to a specific network, and can be re-activated on different LoRaWAN network through OTAA procedure.

Both activation modes require following identifiers & keys configuration:

Mode	ID/EUI	Key
ABP	DevAddr	NwkSKey, AppSKey
OTAA	AppEUI, DevEUI	AppKey

Note: the generation of DevAddr in ABP mode is dependant on the LoRaWAN service provider. It can be any random value generated by the user, or can be generated from EUI (or other mean) by the LoRaWAN service provider.

8.3 Deploying TBSL1/RFB-LoRa on the field

It is key having the gateway installed as high as possible to get increased communication range.

Each node should be carefully configured depending on how far it is from the LoRaWAN gateway, and if there's any obstacle between both (building, trees, hills, etc…):

- Initial data rate should be configured accordingly (warning: user must ensure the maximum payload for the selected data rate matches with TBSL1 packets size. This warning must also be considered in case ADR is enabled). The lowest the data rate is, the more robust is the transmission, but this has an impact on the transmission air time and on the maximum payload.
- Decide whether to use confirmed or unconfirmed messages (rely on LoRaWAN repeat feature then to improve successful reception rate).

9 MQTT network basics

The following introduction of MQTT comes from February 2014 Eclipse newsletter:

MQTT is a publish/subscribe messaging protocol designed for lightweight M2M communications. It was originally developed by IBM and is now an open standard.

9.1 Architecture

MQTT has a client/server model, where every sensor is a client and connects to a server, known as a broker, over TCP.

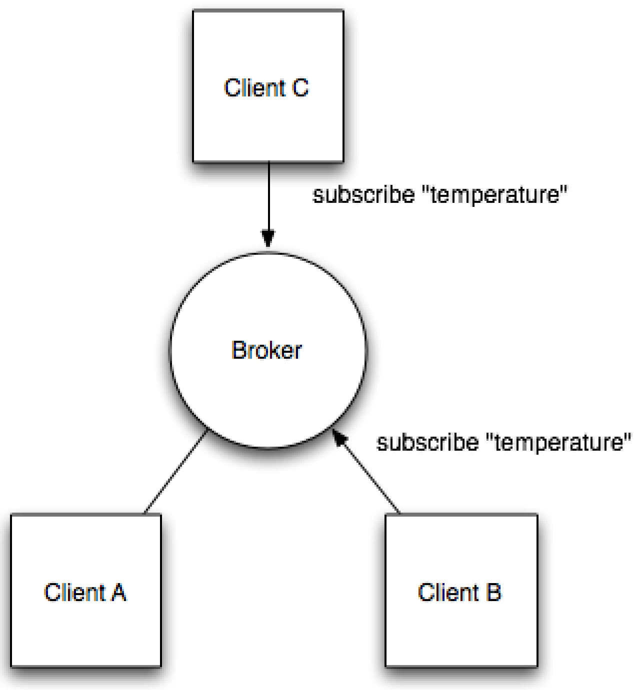
MQTT is message oriented. Every message is a discrete chunk of data, opaque to the broker.

Every message is published to an address, known as a topic. Clients may subscribe to multiple topics. Every client subscribed to a topic receives every message published to the topic.

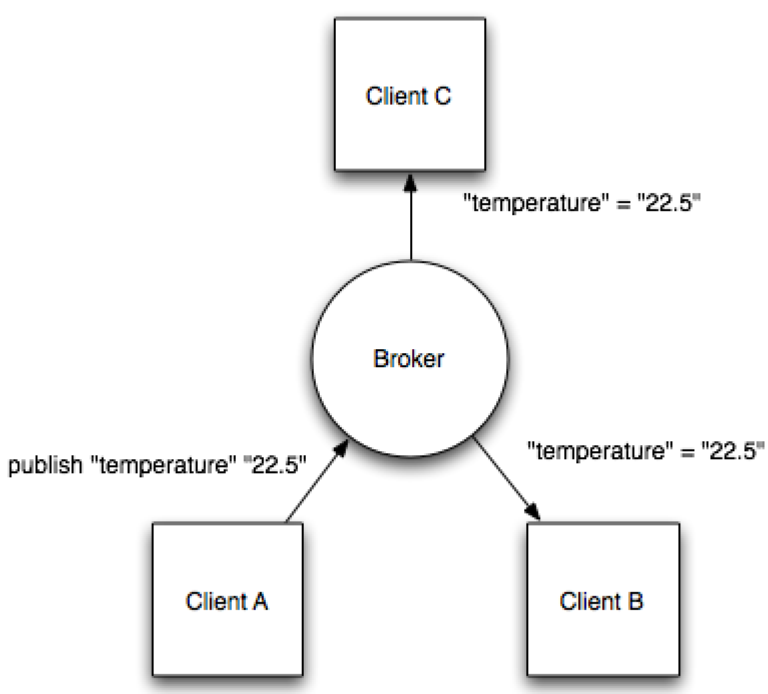
For example, imagine a simple network with three clients and a central broker.

All three clients open TCP connections with the broker. Clients B and C subscribe to the topic temperature .

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com



At a later time, Client A publishes a value of 22.5 for topic temperature . The broker forwards the message to all subscribed clients.



The publisher subscriber model allows MQTT clients to communicate one-to-one, one-to-many and many-to-one.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

9.2 Topic matching

In MQTT, topics are hierarchical, like a filing system (eg. Kitchen/oven/temperature). Wildcards are allowed when registering a subscription (but not when publishing) allowing whole hierarchies to be observed by clients.

The wildcard + matches any single directory name, # matches any number of directories of any name.

For example, the topic kitchen/+/temperature matches kitchen/foo/temperature but not kitchen/foo/bar/temperature

kitchen/# matches kitchen/fridge/compressor/valve1/temperature

9.3 Application Level QoS

MQTT supports three quality of service levels, “Fire and forget”, “delivered at least once” and “delivered exactly once”.

9.4 Last Will And Testament

MQTT clients can register a custom “last will and testament” message to be sent by the broker if they disconnect. These messages can be used to signal to subscribers when a device disconnects.

9.5 Persistence

MQTT has support for persistent messages stored on the broker. When publishing messages, clients may request that the broker persists the message. Only the most recent persistent message is stored. When a client subscribes to a topic, any persisted message will be sent to the client.

Unlike a message queue, MQTT brokers do not allow persisted messages to back up inside the server.

9.6 Security

MQTT brokers may require username and password authentication from clients to connect. To ensure privacy, the TCP connection may be encrypted with SSL/TLS.

10 Environmental Specifications

Symbol	Parameter	Conditions	Min	Max	Unit
T_A	Operating Ambient Temperature Range		-40	+85	°C
T_{STG}	Storage Temperature Range		-40	+85	°C
	Humidity level	Ta=60°C; no condensation	-	95	% R.H

Table 1 – Environmental Specifications

11 ESD Safety

The TBSL1 is a static-sensitive electronic device. Do not operate or store near strong electrostatic fields. Follow guidelines as per EIA/JESD22-A115-A.

12 RoHS Compliance

TBSL1 modules are compliant with the European Union Directive 2002/95/EC Restriction on the Use of Hazardous Substances (RoHS). All designated products have Pb-free terminals.

13 Ordering Information

Part Number	**Description**
TBSL1/RFB-LoRa	LoRaWAN Multi-sensors bridge
TBSL1/RFB-4GWW	GSM/WCDMA/LTE Multi-sensors bridge (global)
TBSL1/RFB-NbIOT-CatM1	NB2/Cat M1 Multi-sensors bridge (global)
TBSL1/RFB-Wifi	Wi-Fi Multi-sensors bridge
*TBSL1/RFB-nnnn-**DC***	TBSL1 variant with external power supply support (nnnn: LoRa, 4GWW, NbIOT-CatM1, Wifi)
TBSSP5W	External 5W solar panel
TBSBR02	Mounting brackets
LLT-M14CM05	5 pins female connector for external solar panel / External power supply
LLT-M14CM07	7 pins female connector for sensors
Not sold by Tekbox *Ex: GEB585460 3.7V*	Rechargeable battery with 10k NTC, protection circuit and 1500-2500mAh capacity

Contact: sales@tekbox.com

14 History

Version	Date	Author	Changes
V1.0	28.04.2017	Philippe Hervieu	Creation of the document
V1.1	24.05.2017	Philippe Hervieu	Update system and power sections
V1.2	29.05.2017	Philippe Hervieu	Typo/spelling in 3.3.3
V1.3	15.09.2017	Philippe Hervieu	Update TBSL1 features description

V1.4	29.09.2017	Philippe Hervieu	Roadmap updated + minor fixes
V1.5	13.11.2017	Philippe Hervieu	Update of LoRaWAN downlink and power management chapters.
V1.6	18.01.2018	Philippe Hervieu	Update Electrical Characteristics section (battery and solar panel)
V1.7	09.03.2018	Philippe Hervieu	Update document with cellular products.
V1.8	10.04.2018	Philippe Hervieu	Update products names
V1.9	25.06.2018	Philippe Hervieu	Update connectivity information, pictures and power management section.
V1.10	30.07.2018	Philippe Hervieu	Analog and pulse sensors electrical requirements
V1.11	23.08.2018	Philippe Hervieu	Update modem information and downlink data format
V1.12	14.09.2018	Philippe Hervieu	Update common report message format / Add battery message format.
V1.13	24.09.2018	Philippe Hervieu	Missing board status ID in common message format
V1.14	02.10.2018	Philippe Hervieu	Limitations related to platform's configuration
V1.15	18.10.2018	Philippe Hervieu	Sensors connectors part numbers and 4G US version
V1.16	14.11.2018	Philippe Hervieu	Update PS message format
V1.17	03.12.2018	Philippe Hervieu	Describe downlink commands responses
V1.18	21.12.2018	Philippe Hervieu	Firmware/MIDlet/certificates over the air update
V1.19	26.12.2018	Philippe Hervieu	Security parameters updated based on MQTT client ID
V1.20	26.02.2019	Philippe Hervieu	Update PA and PP messages payload description
V1.21	26.05.2020	Philippe Hervieu	Update analog jumpers section + TBSL1 cellular variant name.
V1.22	23.09.2020	Philippe Hervieu	Rework of document structure / Pictures updated and additional modem information
V1.23	28.09.2020	Philippe Hervieu	Update pictures in 2.2.1
V1.24	08.03.2020	Philippe Hervieu	Fix comment on LoRa Rf connector, update some references.
V1.25	08.06.2021	Philippe Hervieu	Add chapter related to payload parsing with Javascript regex
V1.26	19.11.2021	Philippe Hervieu	Update C packet format with time stamp.
V1.27	28.02.2022	Philippe Hervieu	Note about retaining real time / Tekbox broker removed / Current limits updated.
V1.28	10.09.2022	Philippe Hervieu	Update 4G module related chapters / Add Wi-Fi and NbIoT/Cat M1 modem support Network time synchronization
V1.29	08.09.2022	Philippe Hervieu	Few updates related to Wi-Fi modem / Update LoRaWAN version.
V1.30	10.03.2023	Philippe Hervieu	Battery charging at negative temperatures – ordering information.
V1.31	15.03.2023	Philippe Hervieu	Add chapter related to battery charging protection.

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com

15 Acronyms

Acronym	Description
ABP	Activation by Personalization
LoRa	Long Range
LTE	Long Term Evolution
MQTT	Message Queuing Telemetry Transport
NC	Not Connected
NTP	Network Time Protocol
OTAA	Over The Air Activation
QoS	Quality of Service
RAT	Radio Access Technology
RFU	Reserved for Future Use
RTC	Real Time Clock

Factory 4, F4, Lot I-3B-1 | Saigon Hi-Tech Park | Tan Phu Ward, District 9 | Ho Chi Minh City | Tel +84 (83)5471340
E-mail office@tekbox.com| www.tekbox.com