V1.5

User Guide TBSL1ConfigurationTool

PC GUI for TBSL1 configuration

Table of Contents

1 Introduction

TBSL1ConfigurationTool is .NET-based utility program that can interface to TBSL1 through USB (Serial COM Port) connection. It can be used as a tool to set up operational parameters: timings, sensors settings (SDI-12, pulse and analog sensors) and modem configuration (LoRa/3G/4G).
The GUI can also be used as a test and diagnostic tool as it allows to send commands directly to SDI-12 sensors (according to SDI-12 standard) and LoRaWan/Cellular modem (through AT commands) and display their responses.

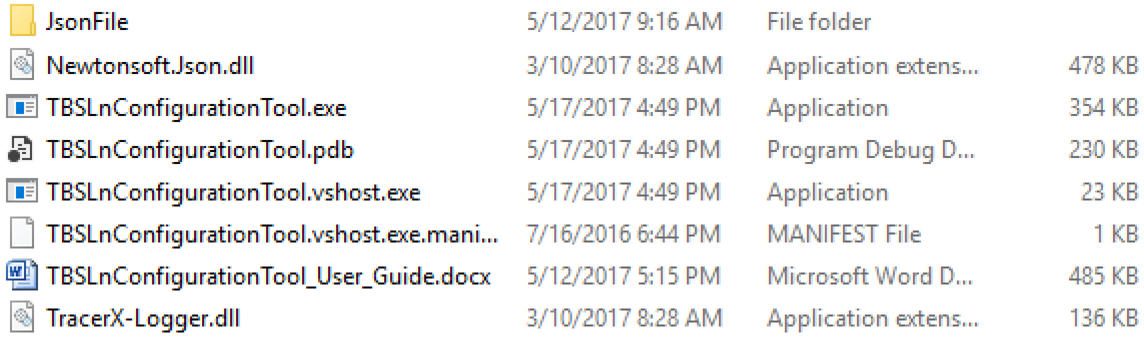
2 FW Upgrade

Whenever user update a new version of TBSL1 FW or GUI make sure to switch both platform and GUI to Console Mode before performing an update.

Check 4.7 Operation notification for more details.

3 Software dependencies and application's files

TBSL1ConfigurationTool is a .NET application developed in C# language with .NET Framework 4. So, to able to run the application, it is required to have .NET Framework 4 package installed on the target PC.
Basically, the application consists of some files and one folder which contain 2 JSON files. While operating, it will automatically create "Log" directory at current application location to store logging information and errors as well.

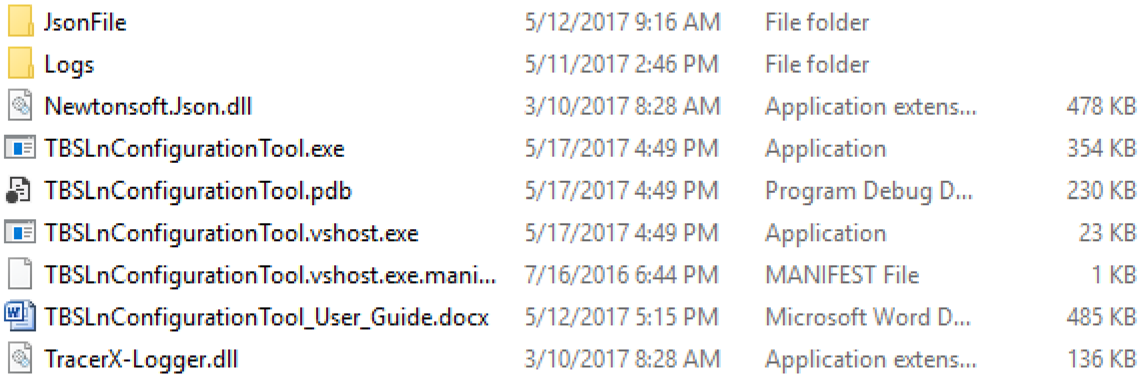


JsonFile	5/12/2017 9:16 AM	File folder	
Newtonsoft.Json.dll	3/10/2017 8:28 AM	Application extens...	478 KB
TBSLnConfigurationTool.exe	5/17/2017 4:49 PM	Application	354 KB
TBSLnConfigurationTool.pdb	5/17/2017 4:49 PM	Program Debug D...	230 KB
TBSLnConfigurationTool.vshost.exe	5/17/2017 4:49 PM	Application	23 KB
TBSLnConfigurationTool.vshost.exe.mani...	7/16/2016 6:44 PM	MANIFEST File	1 KB
TBSLnConfigurationTool_User_Guide.docx	5/12/2017 5:15 PM	Microsoft Word D...	485 KB
TracerX-Logger.dll	3/10/2017 8:28 AM	Application extens...	136 KB

Fig.1 Initial Application Files.

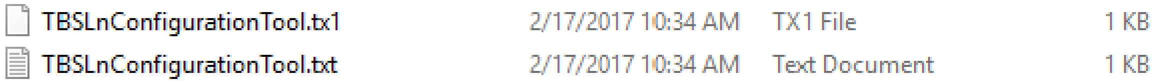
+ Newtonsoft.Json.dll – DLL file to support JSON format.
+ TBSL1ConfigurationTool.exe – main application file. To run the application, we execute this file.
+ TBSL1ConfigurationTool_User_Guide.docx – word file contain instruction for user how to use the application.
+ TraceX-Logger.dll – DLL file to support logging functions.
+ JsonFile folder – the folder contain 2 JSON file and a configuration folder:

- Settings.json – JSON file which is used to hold all parameters of the application and load it when start the GUI. This always hold the latest setting even if user save their parameter in other JSON file.
- DefaultSettings.json – optional; used to restore default settings.
- ReadOnly/CurrentMode.json – must not be modified by the user, contains the GUI operating mode (console or logging).



JsonFile	5/12/2017 9:16 AM	File folder	
Logs	5/11/2017 2:46 PM	File folder	
Newtonsoft.Json.dll	3/10/2017 8:28 AM	Application extens...	478 KB
TBSLnConfigurationTool.exe	5/17/2017 4:49 PM	Application	354 KB
TBSLnConfigurationTool.pdb	5/17/2017 4:49 PM	Program Debug D...	230 KB
TBSLnConfigurationTool.vshost.exe	5/17/2017 4:49 PM	Application	23 KB
TBSLnConfigurationTool.vshost.exe.mani...	7/16/2016 6:44 PM	MANIFEST File	1 KB
TBSLnConfigurationTool_User_Guide.docx	5/12/2017 5:15 PM	Microsoft Word D...	485 KB
TracerX-Logger.dll	3/10/2017 8:28 AM	Application extens...	136 KB

Fig.2 Directory Structure when Application Running.

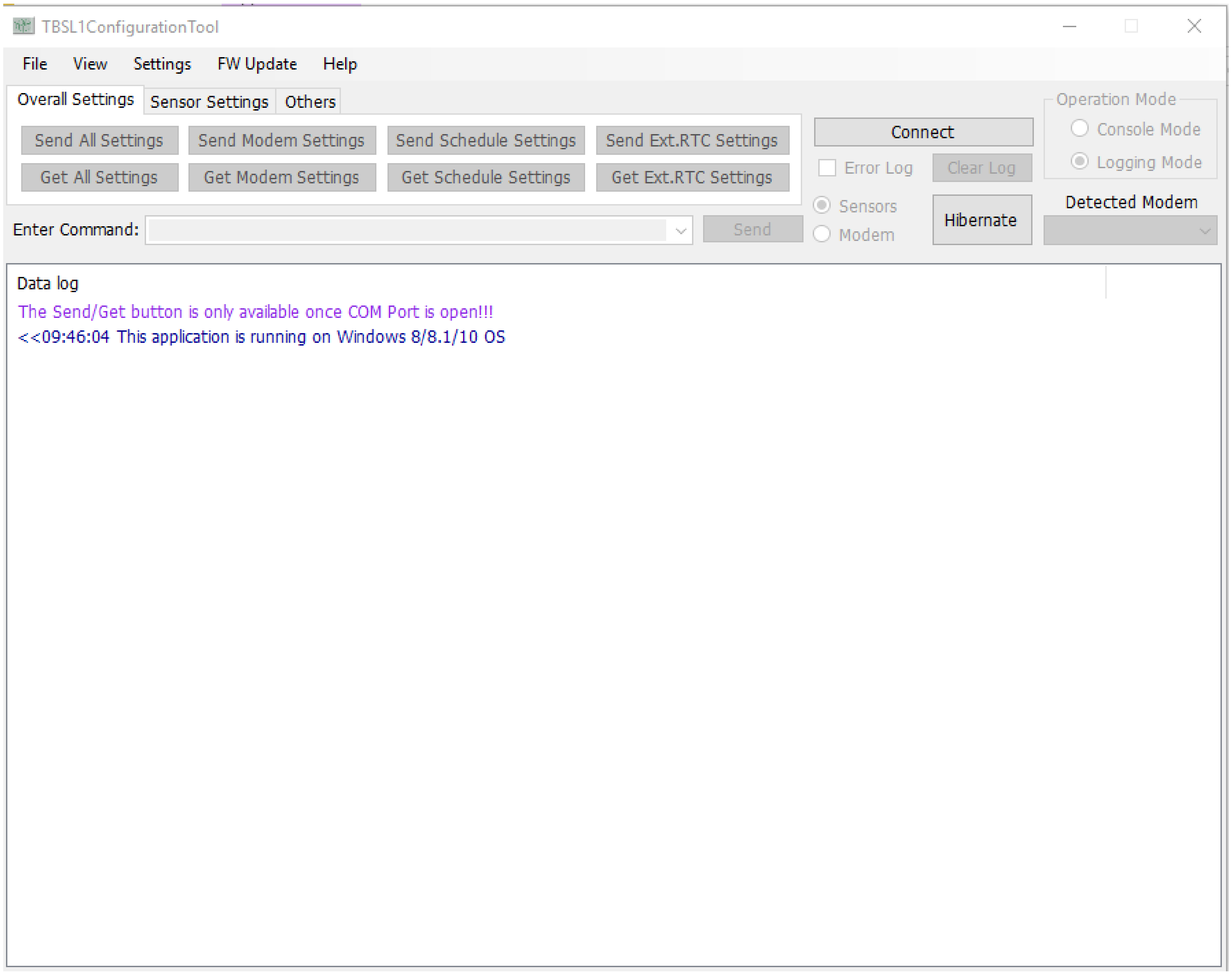


TBSLnConfigurationTool.tx1	2/17/2017 10:34 AM	TX1 File	1 KB
TBSLnConfigurationTool.txt	2/17/2017 10:34 AM	Text Document	1 KB

Fig.3 Log File.

4 Functional Description

To open the GUI application, execute file TBSL1ConfigurationTool.exe with administrator rights in the directory where the application files are located.



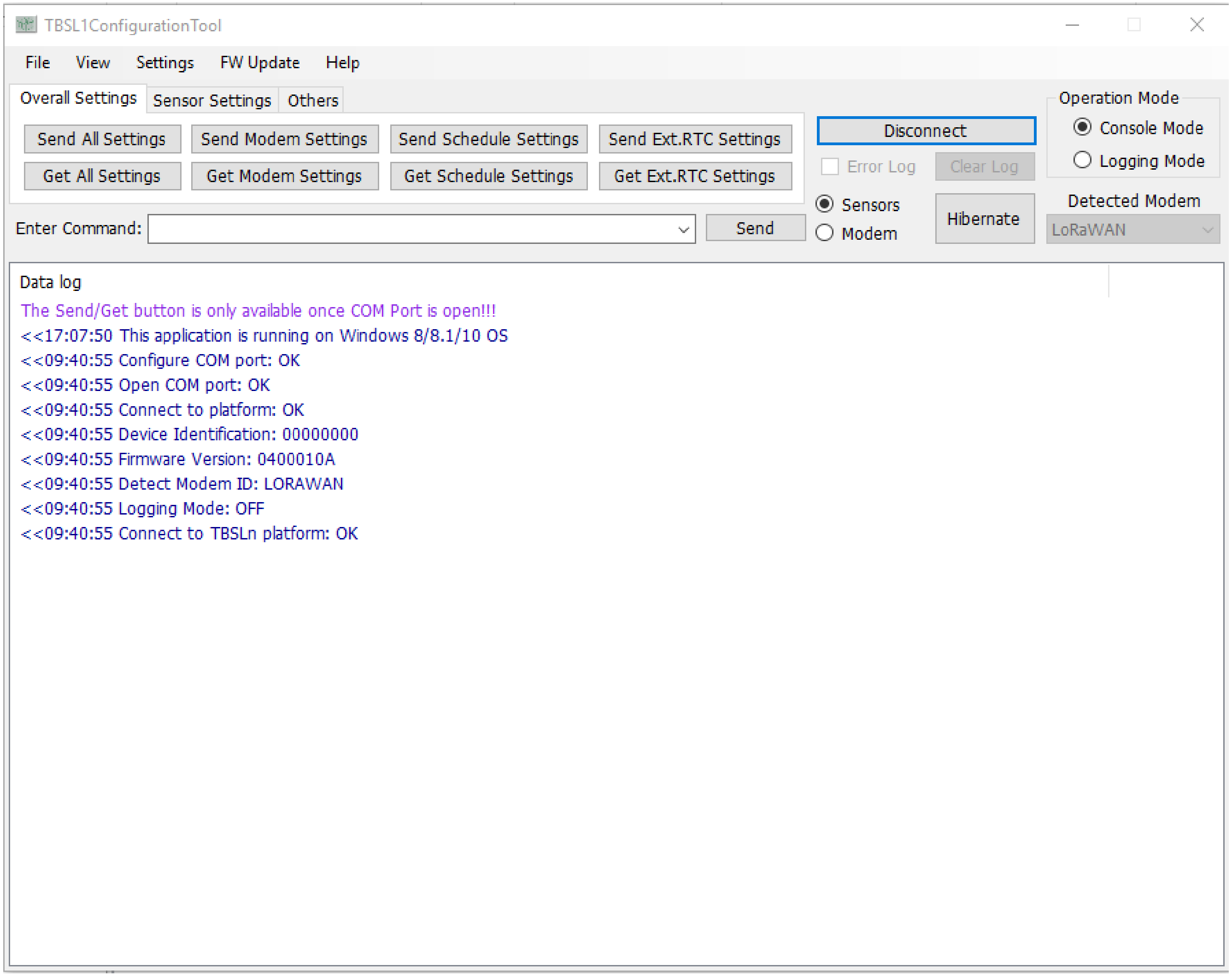
```
Data log
The Send/Get button is only available once COM Port is open!!!
<<09:46:04 This application is running on Windows 8/8.1/10 OS
```

Fig.4 Main Application Screen.

When the GUI has started, it shows the current mode of platform, if the GUI opened for the first time the current mode is always Console Mode. After that, it depends on what mode was the platform before closing the GUI.

Before COM Port is opened, almost all buttons will be disabled or will not work.

After COM Port is opened, the GUI will check connection with the platform and automatically get external RTC time from the platform and display it on the GUI.
Note:Sometimes the displayed External RTC Time differs by one second from PC time. This is only a display lagging issue that doesn't affect the platform.



```
The Send/Get button is only available once COM Port is open!!!
<<17:07:50 This application is running on Windows 8/8.1/10 OS
<<09:40:55 Configure COM port: OK
<<09:40:55 Open COM port: OK
<<09:40:55 Connect to platform: OK
<<09:40:55 Device Identification: 00000000
<<09:40:55 Firmware Version: 0400010A
<<09:40:55 Detect Modem ID: LORAWAN
<<09:40:55 Logging Mode: OFF
<<09:40:55 Connect to TBSLn platform: OK
```

Fig.5 GUI after opening COM Port.

4.1 File Menu

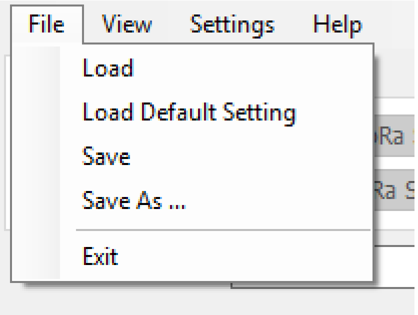


Fig.6 File Menu.

- **“Load” item:** this item allows user to load setting from JSON file. After clicking on this item, a file dialog window will be open for user to choose suitable JSON file.

- **"Load Default Setting" item:** this item allows user to load setting from DefaultSetting.json file. This file contain default settings for all parameter in the GUI.
- **"Save" item**: this item allows user to save setting to default JSON file "Settings.JSON". The settings stored in "Settings.JSON" file usually are the newest settings user had set, it will be loaded each time the GUI starts up.
- **"Save As …" item**: this item allows user to save setting to any JSON file they want to.
- **"Exit" item**: Stop and quit the application.

4.2 View Menu

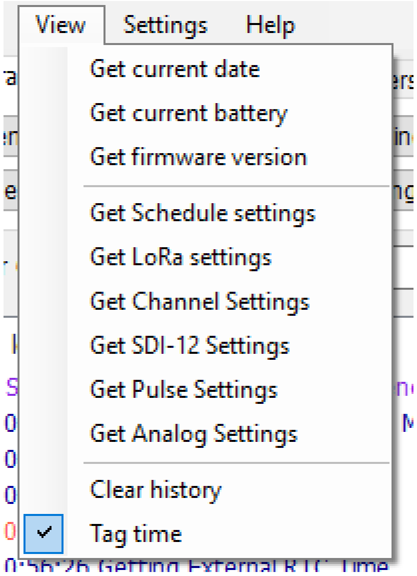


Fig.7 TBSL1 Menu.

- "Get current date" item: When the bridge operates in console mode, this item allows to get the platform's current time. It is displayed as below on GUI when this item is clicked**.**

- "Get current battery" item: When the bridge operates in console mode, this item allows to get the battery voltage. It is displayed as below on GUI when this item is clicked.

- "Get firmware version" item: When the bridge operates in console mode, this item allows to get the version of current firmware. The firmware version string has format: <Model><Custom><Version><Revision>, where:

- Model: 04 for TBSL1
- Custom: used for debug FW, custom features, etc…
- Version: FW version
- Revision: FW revision
- Example: 0400010A

- "Get Schedule settings" item: When the bridge operates in console mode, this item allows to get the current schedule settings on the bridge such as: all kind of sensors measurement interval, transmission interval, battery sending cycle. Schedule parameters are returned to the GUI when this item is clicked.

- "Get LoRa settings" item: When the bridge operates in console mode, this item allows to get current LoRa settings on the bridge. Lora parameters are displayed on GUI when we click on this item

- "Get Channel Settings" item: When the bridge operates in console mode, this item allows to get current Channel settings on the bridge. All settings for 8 channels (even disabled channel) are displayed on the GUI when this item is clicked (Note: when frequency is set to 0, it means corresponding channel is disabled). (RHF LoRaWAN 2.0.10)(different in RHF LoRaWAN 2.1.x)

- "Get SDI-12 Settings" item: When the bridge operates in console mode, this item allows to get SDI-12 command list on the bridge. SDI-12 commands list is displayed on GUI when this item is clicked. Each command displayed on each row consists in two sub-commands. A command is used to measure and the other for getting data.

- "Get Pulse Settings" button: When the platform operates in console mode, this button is used to get Pulse sensor settings from the platform.

- "Get Analog Settings" button: When the platform operates in console mode, this button is used to get Analog sensor settings from the platform.

- "Clear history" item: Clear all information that was displayed on Data log window.

- "Tag time" item: Start-up state of this item was selected. When the item is selected, the application will tag PC's current time to every data that it sends/receives to/from the bridge. This feature is useful when we need to calculate time stamps for commands.

4.3 Settings Menu

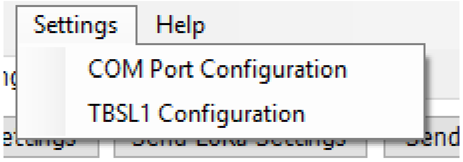
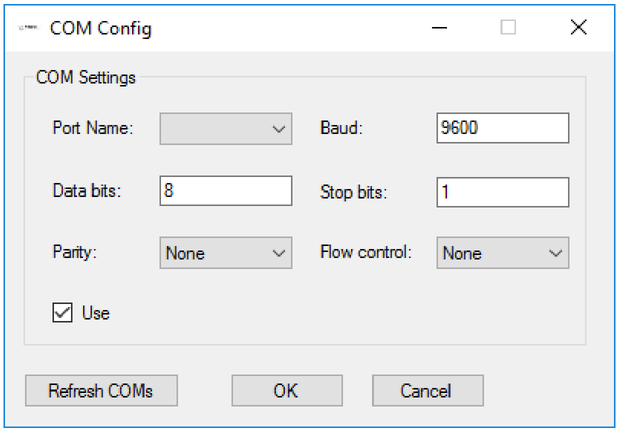


Fig.8 Settings Menu.

4.3.1 COM Port Configuration

- “COM Port Configuration” item: The item is used to configure settings of COM port that interfaces with the platform. When clicked, the application will open new COM settings window such as:



COM Config

COM Settings

Port Name:

Baud: 9600

Data bits: 8

Stop bits: 1

Parity: None

Flow control: None

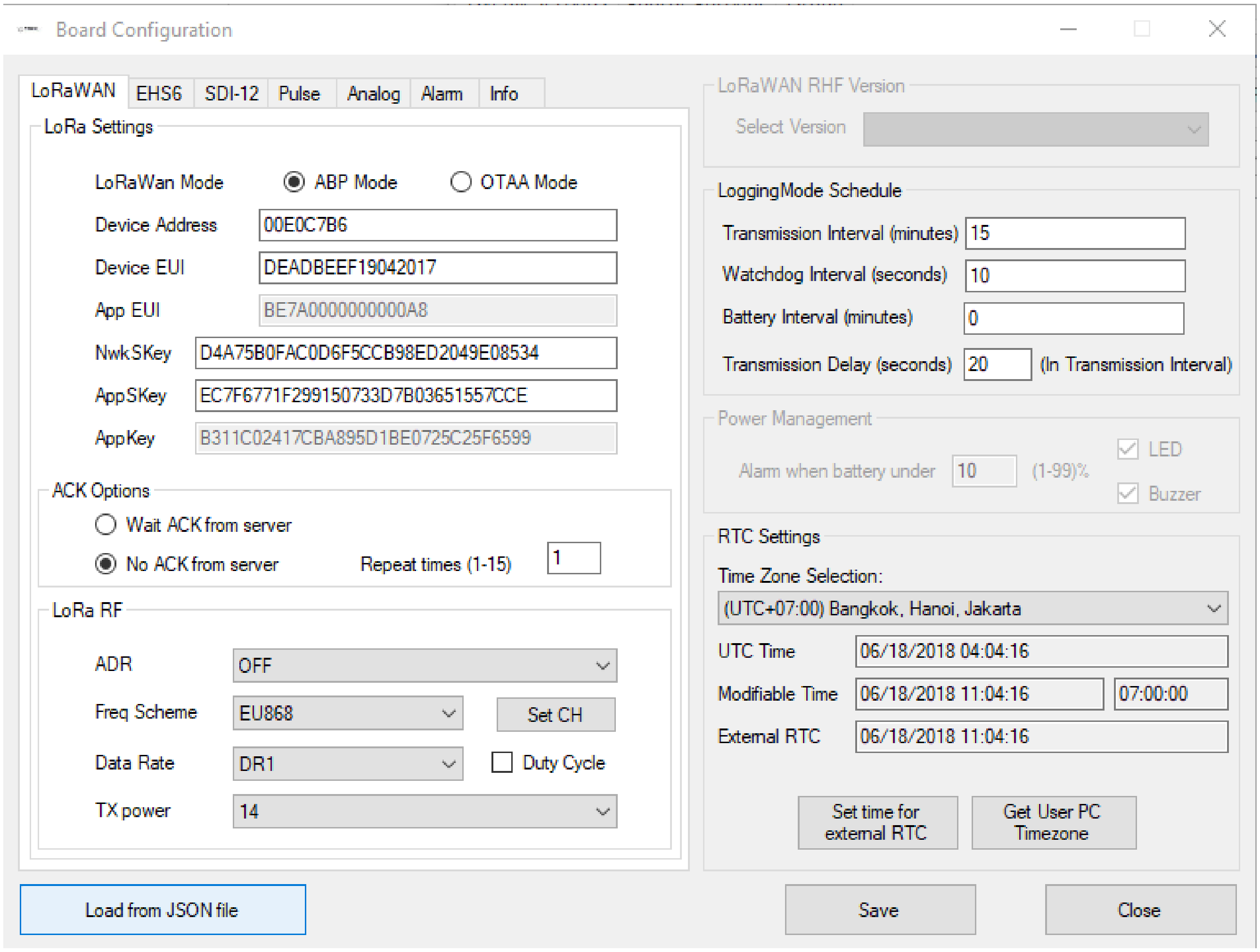
☑ Use

Fig.9 COM Settings.

Correct COM port from “Port Name” drop-list must be selected. If the COM/USB cable is plugged on the PC after opening the window, “Refresh COMs” button can be used to refresh the drop-list. The remaining settings should be kept as default values unless it is needed these changes match with new firmware parameters on the platform. Click “OK” button to accept and save selected values. The new settings will be updated and written to the appropriate section in the Settings.JSON file.

4.3.2 TBSLn Configuration

- “TBSLn Configuration” item: The item is used to configure settings for the platform . When clicked, the application will open a new window such as:



Board Configuration

LoRaWAN | EHS6 | SDI-12 | Pulse | Analog | Alarm | Info

LoRa Settings

LoRaWan Mode ● ABP Mode ○ OTAA Mode

Device Address: 00E0C7B6

Device EUI: DEADBEEF19042017

App EUI: BE7A0000000000A8

NwkSKey: D4A75B0FAC0D6F5CCB98ED2049E08534

AppSKey: EC7F6771F299150733D7B03651557CCE

AppKey: B311C02417CBA895D1BE0725C25F6599

ACK Options

○ Wait ACK from server

● No ACK from server Repeat times (1-15): 1

LoRa RF

ADR: OFF

Freq Scheme: EU868 [Set CH]

Data Rate: DR1 □ Duty Cycle

TX power: 14

LoRaWAN RHF Version

Select Version:

LoggingMode Schedule

Transmission Interval (minutes): 15

Watchdog Interval (seconds): 10

Battery Interval (minutes): 0

Transmission Delay (seconds): 20 (In Transmission Interval)

Power Management

Alarm when battery under: 10 (1-99)% ☑ LED ☑ Buzzer

RTC Settings

Time Zone Selection: (UTC+07:00) Bangkok, Hanoi, Jakarta

UTC Time: 06/18/2018 04:04:16

Modifiable Time: 06/18/2018 11:04:16 07:00:00

External RTC: 06/18/2018 11:04:16

[Set time for external RTC] [Get User PC Timezone]

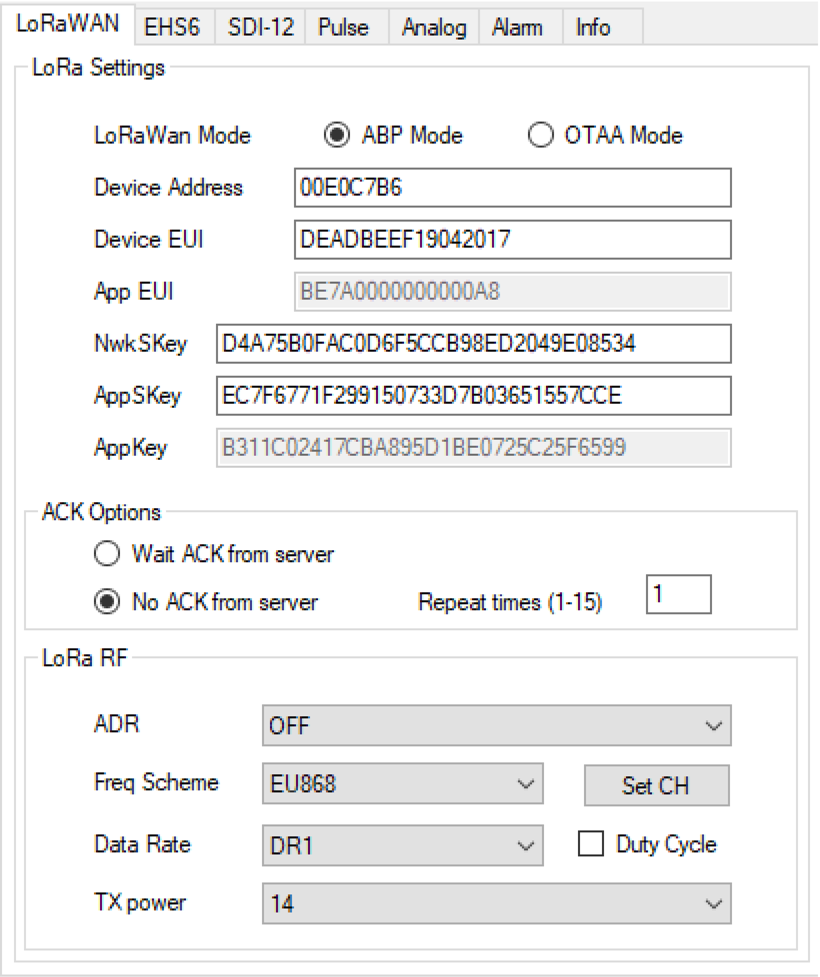
[Load from JSON file] [Save] [Close]

Fig.10 Platform Configuration.

On this window, we have two areas:

a) On the left side is an area for configuring the modem (LoRaWan/Cellular) and 3 kind of sensors (SDI-12, Pulse, Analog). Concerning the modem, only the settings of the detected modem are accessible while others are not (e.g if LoRaWAN modem has been detected by the system then the cellular modem configuration won't be usable).

   i) "LoRaWan" tab: this tab contains all required settings to support LoRaWAN communication:



LoRa Settings

LoRaWan Mode ● ABP Mode ○ OTAA Mode

Device Address: 00E0C7B6

Device EUI: DEADBEEF19042017

App EUI: BE7A0000000000A8

NwkSKey: D4A75B0FAC0D6F5CCB98ED2049E08534

AppSKey: EC7F6771F299150733D7B03651557CCE

AppKey: B311C02417CBA895D1BE0725C25F6599

ACK Options

○ Wait ACK from server

● No ACK from server Repeat times (1-15): 1

LoRa RF

ADR: OFF

Freq Scheme: EU868

Data Rate: DR1 □ Duty Cycle

TX power: 14

Fig.11 LoRaWAN Settings.

(1) "LoRaWan Mode": set activation mode for the LoRa module, either ABP or OTAA.

(2) "Device Address": set address for the LoRa module.

(3) "Device EUI": set Device EUI for the LoRa module.

(4) "App EUI": set Application EUI for the LoRa module.

(5) "NwkSKey": set Network Session Key for the LoRa module.

(6) "AppSKey": set Application Session Key for the LoRa module.

(7) "AppKey": set Application Key for the LoRa module.

(8) "Wait ACK from server": send packet to server and request an acknowledge from the server (LoRaWAN confirmed message)

(9) "No ACK from server": send packet to the server without requesting an acknowledge from the server (LoRaWAN unconfirmed message)

- Repeat time(1-15): number of retransmission of the packet, this is handled automatically by the modem (refer to LoRaWAN standard).

(10) "ADR": set ADR (LoRaWAN Adaptive Data Rate), user can choose between "ON" and "OFF".

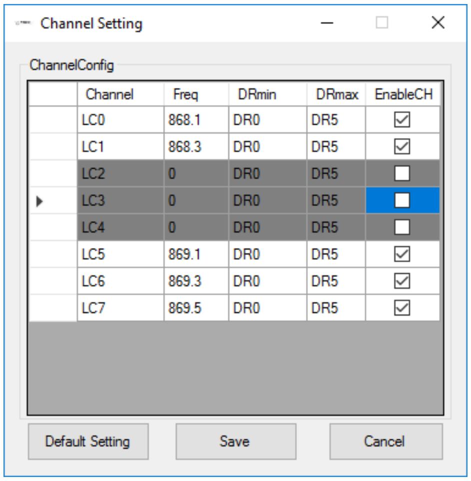
(11) “Freq Scheme”: set datarate scheme for LoRa depending on user’s region. The application supports EU868, US915, US915HYBRID, AU915, EU433, AS923, CN779 and CN470 for now. Other regions might be supported in the future.

(12) “Data Rate”: set (initial) datarate for LoRa, the available range depends on chosen “Freq Scheme”. Ex: In EU868, “Data Rate” options will be available between DR0 - DR5.

(13) “Duty Cycle”: this option will only be available in EU868.

(14) “TX power”: these item values also depend on which “Freq Scheme” is chosen.

(15) “Set CH”: settings for 8 channels. The below configuration screen displays default channel settings: user can only enable/disable channels or set frequencies and channel numbers.



	Channel	Freq	DRmin	DRmax	EnableCH
	LC0	868.1	DR0	DR5	☑
	LC1	868.3	DR0	DR5	☑
	LC2	0	DR0	DR5	☐
▸	LC3	0	DR0	DR5	☐
	LC4	0	DR0	DR5	☐
	LC5	869.1	DR0	DR5	☑
	LC6	869.3	DR0	DR5	☑
	LC7	869.5	DR0	DR5	☑

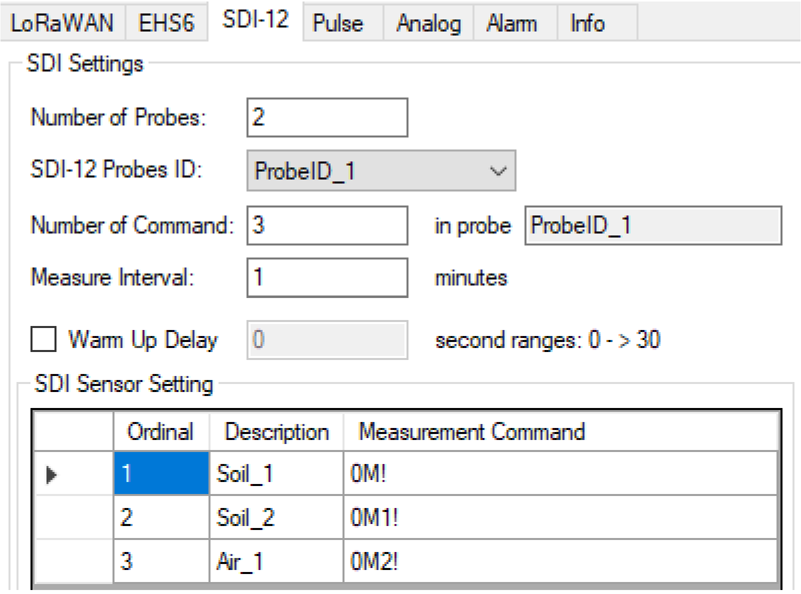
Fig.12 Channel Settings.

DRmin and DRmax can’t be edited.
User can disable channel by unchecking EnableCH or deleting the Frequency. After disabling a channel, user can only enable it again by clicking on EnableCH and setting the frequency.

- “Default Settings” button: set all settings back to default.
- “Save” button: save user settings. To save to JSON file, user must also click “Save”/”Save as” button from “File” drop down menu.
- “Cancel” button: discards modified channel settings.

ii) “CELLULAR” tab: used to configure cellular modem (3G, 4G) and set MQTT parameters.

iii) “SDI-12 Sensor” tab:



SDI Settings

Number of Probes: 2

SDI-12 Probes ID: ProbeID_1

Number of Command: 3 in probe ProbeID_1

Measure Interval: 1 minutes

□ Warm Up Delay 0 second ranges: 0 - > 30

SDI Sensor Setting

	Ordinal	Description	Measurement Command
▶	1	Soil_1	0M!
	2	Soil_2	0M1!
	3	Air_1	0M2!

Fig.13 SDI-12 Sensor Settings.

(1) "Number of Probes": set the number of SDI-12 probes user will use. If there is no sensor, the user needs to set this parameter first before configuring any sensor.

(2) "SDI-12 Sensor ID": This is a virtual ID starting at 1 and automatically generated by the GUI.

(3) "Number of Command": set the total numbers of SDI-12 commands for selected probes. This matches with number of Ordinal in SDI Command table.

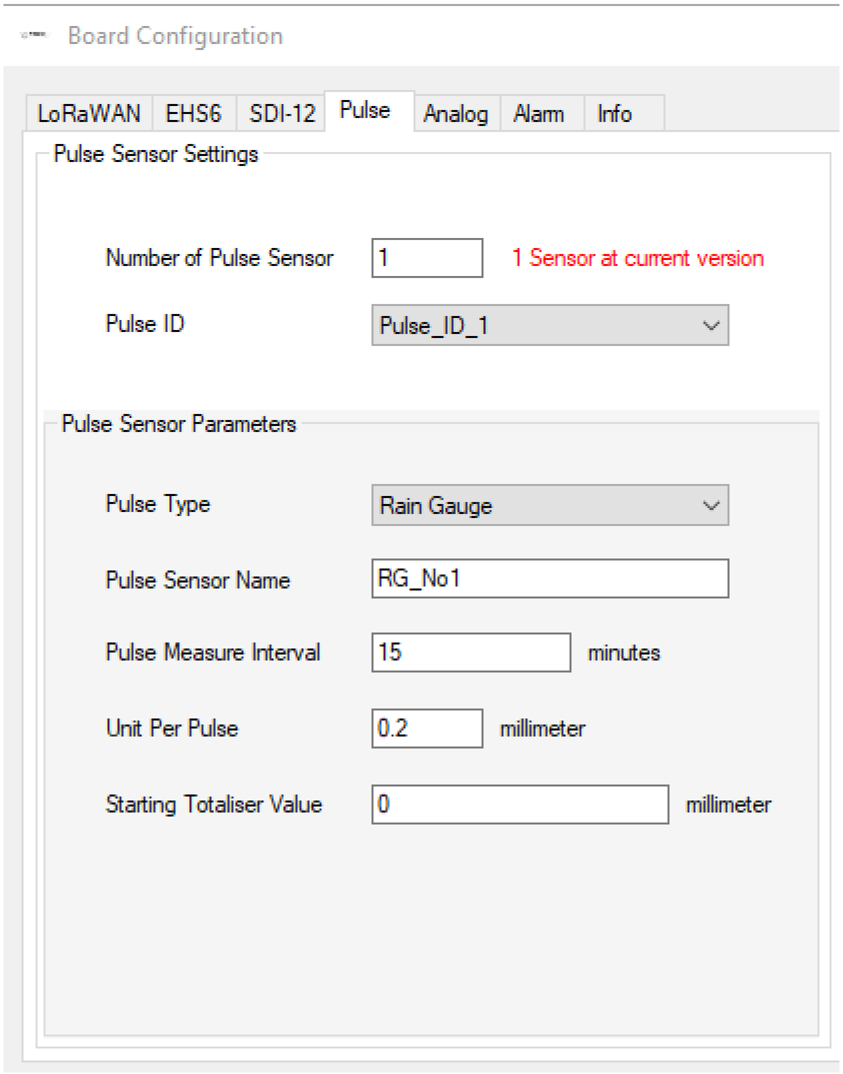
(4) "Measure Interval": set measurement interval (ie logging period) for current SDI-12 probe ID.

(5) "Warm Up Delay": for SDI-12 sensors that require a warm up time before measurement.

(6) "SDI Sensor Setting" area: this area contains parameter of SDI-12 Sensor. User must choose Probes ID in order to enable it, "SDI Sensor Setting" area will be disabled if no Sensor ID is selected. Below are the parameters of SDI-Sensor:

- SDI Command table: set Measurement Command and Description for specific SDI-Sensor.
- Ordinal: This is index of each Command in SDI Command table and it is automatically generated by the GUI.

iv) "Pulse Sensor" tab:



Board Configuration

Pulse Sensor Settings

Number of Pulse Sensor: 1 — 1 Sensor at current version

Pulse ID: Pulse_ID_1

Pulse Sensor Parameters

Pulse Type: Rain Gauge

Pulse Sensor Name: RG_No1

Pulse Measure Interval: 15 minutes

Unit Per Pulse: 0.2 millimeter

Starting Totaliser Value: 0 millimeter

Fig.14 Pulse Sensor Settings.

(1) "Number of Pulse Sensor": set the number of Pulse sensors. If there is no sensor, user needs to set this first before configuring any sensor.

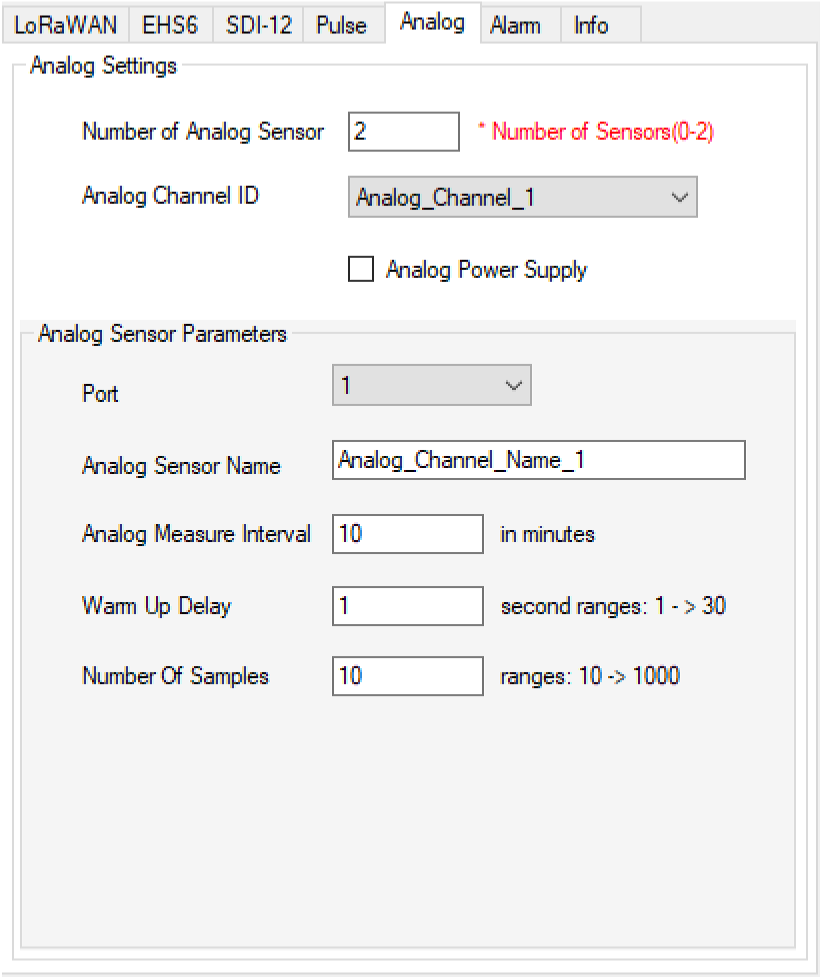
(2) "Pulse Sensor ID": This is a virtual ID starting at 1 and automatically generated by the GUI.

(3) "Pulse type": choose between 2 types of applications, either rain gauge or flow meter.

(4) "Pulse Sensor Parameters" area: this area contains parameters related to pulse sensor. User must choose Sensor ID in order to enable it, "Pulse Sensor Parameters" area will be disabled if no Sensor ID is selected. Below are the parameters of pulse sensor:

- Sensor Name: name of configured pulse sensor, customised by the user.
- "Pulse Measure Interval": set measurement interval for specific pulse sensor.
- "Unit per pulse": set to match the sensor e.g. 0.2mm per pulse for rain gauge.
- "Starting Totalizer Value": when using TBSL1 on a flow meter and it is needed to have the totalizer value to match the value on the meter's mechanical dial, set the totalizer to the current value shown on the dials.

v) "Analog Sensor" tab:



Analog Settings

Number of Analog Sensor: 2 * Number of Sensors(0-2)

Analog Channel ID: Analog_Channel_1

□ Analog Power Supply

Analog Sensor Parameters

Port: 1

Analog Sensor Name: Analog_Channel_Name_1

Analog Measure Interval: 10 in minutes

Warm Up Delay: 1 second ranges: 1 - > 30

Number Of Samples: 10 ranges: 10 -> 1000

Fig.15 Analog Sensor Settings.

(1) "Number of Analog Sensor": set the number of Analog sensors. If there is no sensor, user needs to set this first before configuring any sensor.

(2) "Analog Channel ID": This is a virtual ID starting at 1 and automatically generated by the GUI.

(3) "Analog Power Supply": for analog sensors that need to be permanently powered.

(4) "Analog Sensor Parameters" area: this area contain parameter of Analog sensor. User must choose Sensor ID in order to enable it, "Analog Sensor Parameters" area will be disable if no Sensor ID selected. Belows is the parameter of Analog Sensor:

- "Analog Sensor Name": name of configured analog sensor, customised by the user.
- "Analog Measure Interval": set measurement interval for specific Analog Sensor.
- "Port": set port for specific analog sensor, default port is 1.

vi) "Alarm" tab: create an alarm if a parameter goes outside of preset limits. Alarms can be generated based on the value of any of the SDI-12, pulse or analog sensors.

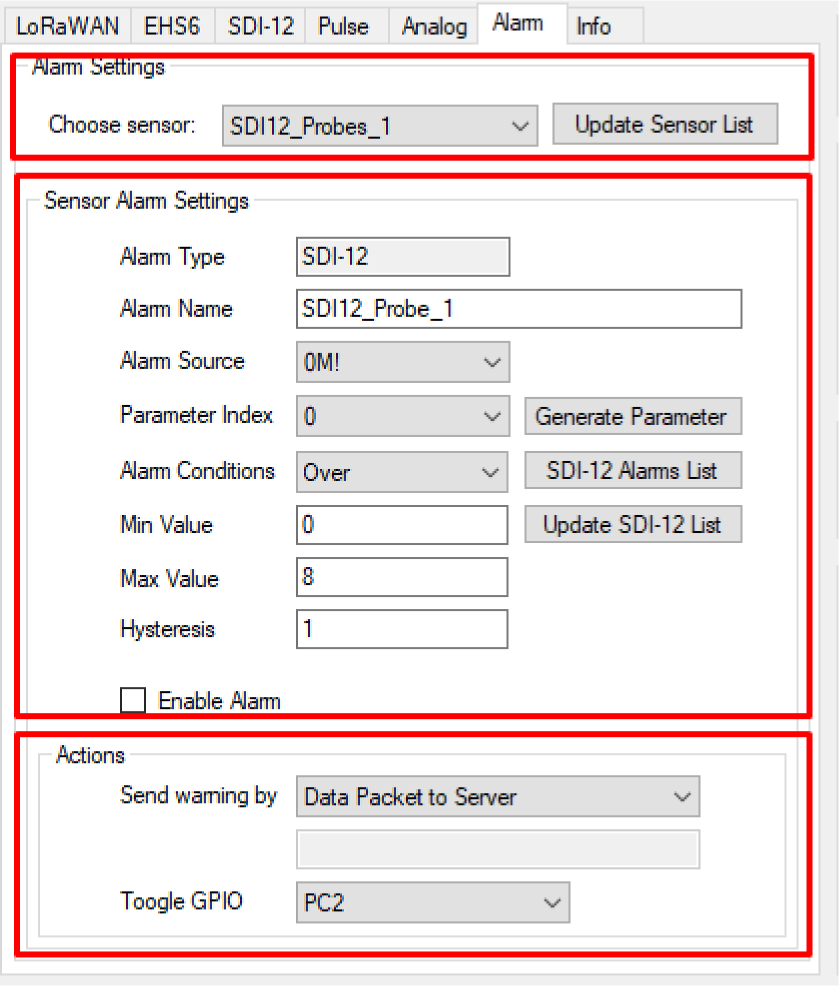
(1) "Choose sensor": select the sensor to which alarm will be applied from the drop down menu. By clicking on "Update sensor list", the list will be populated with all sensors connected to the platform and registered through the GUI.

(2) "Sensors Alarm Settings":

- "Alarm Type": based on selected sensor's type.
- "Alarm Name": based on selected sensor's name.
- "Alarm Source": select which sensor's measurement will be compared to alarm condition (e.g. choose whether the alarm will apply to the temperature or humidity measurement returned by the sensor).
- "Parameter Index": sub-sensor index, applicable to SDI-12 sensors.
- "Alarm Conditions": select alarm condition from the drop down menu, ie under, over, inside or outside limit(s).
- "Min Value": lower threshold to be used by alarm condition.
- "Max Value": upper threshold to be used by alarm condition.
- "Hysteresis": value of hysteresis, set to zero if none. This is the range the value must recover before a new alarm can be triggered and is designed to prevent multiple alarms being generated if a value is sitting close to the alarm point.
- "Enable Alarm": tick this box to enable the alarm for selected sensor.

(3) "Actions": when the alarm is detected, select which action is triggered.

- "Send warning by": data packet to server (LoRa/cellular), SMS (cellular version only), E-mail (cellular version only) or no warning sent.
- "Toggle GPIO": upon alarm, a GPIO from TBSL1 can be triggered (to control an irrigation vanne for instance). Available GPIO are dependant on TBSL1 HW versions.



Alarm Settings

Choose sensor: SDI12_Probes_1

Sensor Alarm Settings

Alarm Type: SDI-12

Alarm Name: SDI12_Probe_1

Alarm Source: 0M!

Parameter Index: 0

Alarm Conditions: Over

Min Value: 0

Max Value: 8

Hysteresis: 1

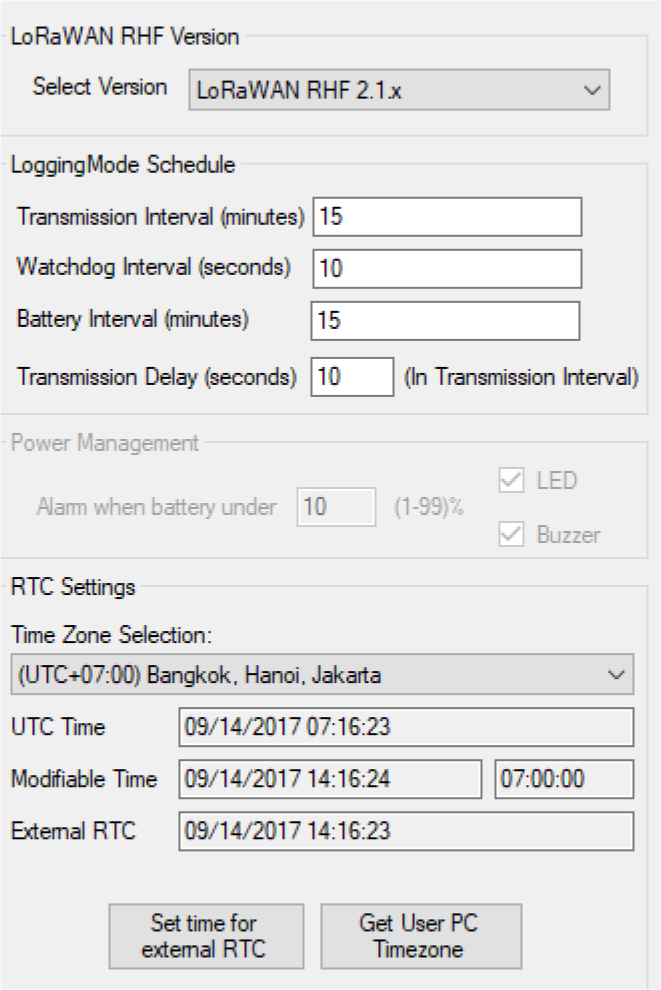
☐ Enable Alarm

Actions

Send warning by: Data Packet to Server

Toogle GPIO: PC2

b) On the right side is an area for settings the platform schedule, power management and date/time for RTC:



LoRaWAN RHF Version

Select Version: LoRaWAN RHF 2.1.x

LoggingMode Schedule

Transmission Interval (minutes): 15

Watchdog Interval (seconds): 10

Battery Interval (minutes): 15

Transmission Delay (seconds): 10 (In Transmission Interval)

Power Management

Alarm when battery under: 10 (1-99)%

☑ LED

☑ Buzzer

RTC Settings

Time Zone Selection: (UTC+07:00) Bangkok, Hanoi, Jakarta

UTC Time: 09/14/2017 07:16:23

Modifiable Time: 09/14/2017 14:16:24 07:00:00

External RTC: 09/14/2017 14:16:23

Fig.16 General Settings.

i) "LoggingMode Schedule" area:

(1) "Transmission Interval (minutes)": period of measurements packets transmission.
(2) "Watchdog Interval (seconds)": (Reserved for Future Use).
(3) "Battery Interval (minutes)": period of reporting for battery and RSSI information.
(4) "Transmission delay (seconds)": offset in seconds based on the transmission interval. Used to stagger the transmission of multiple RTUs using same transmission interval to limit the collisions rate.

ii) "Power Management" area: (disabled in current version)
(1) "Alarm when battery under": Alarm user when platform battery voltage go low.
(2) "LED" checkbox: Enable/disable the LED of platform.
(3) "Buzzer" check box: Enable/disable the Buzzer of platform.

iii) "RTC" area: (time error with current time is < 1 second)

1) “Time Zone Selection”: set time zone for the GUI. The GUI time zone is separate with PC time zone. The GUI time zone is using to set date/time for external and internal.
2) “UTC Time”: Represent the current UTC time. Read only
3) “Local Time”: Represent the current local time and the offset with “UTC Time”. Change depend on “Time Zone Selection”.
4) “External RTC”: Represent the current time of external RTC.
5) “Set time for external RTC” button: set date/time for external RTC. Depend on date/time in “Local Time”.
6) “Get User PC Timezone” button: set “Time Zone Selection”, “Local Time” into current user’s PC timezone.

iv) “LoRaWAN RHF Version”: select LoRa modem FW version. This option should be set to its default value for latest TBSL1 FW (ie “LoRaWAN RHF 2.1.x”, which means the modem is compliant with the latest LoRaWAN standard, version 1.02).

4.4 Sensors

Configuration area for sensors that are TBSL1 variant dependant (e.g digital camera).

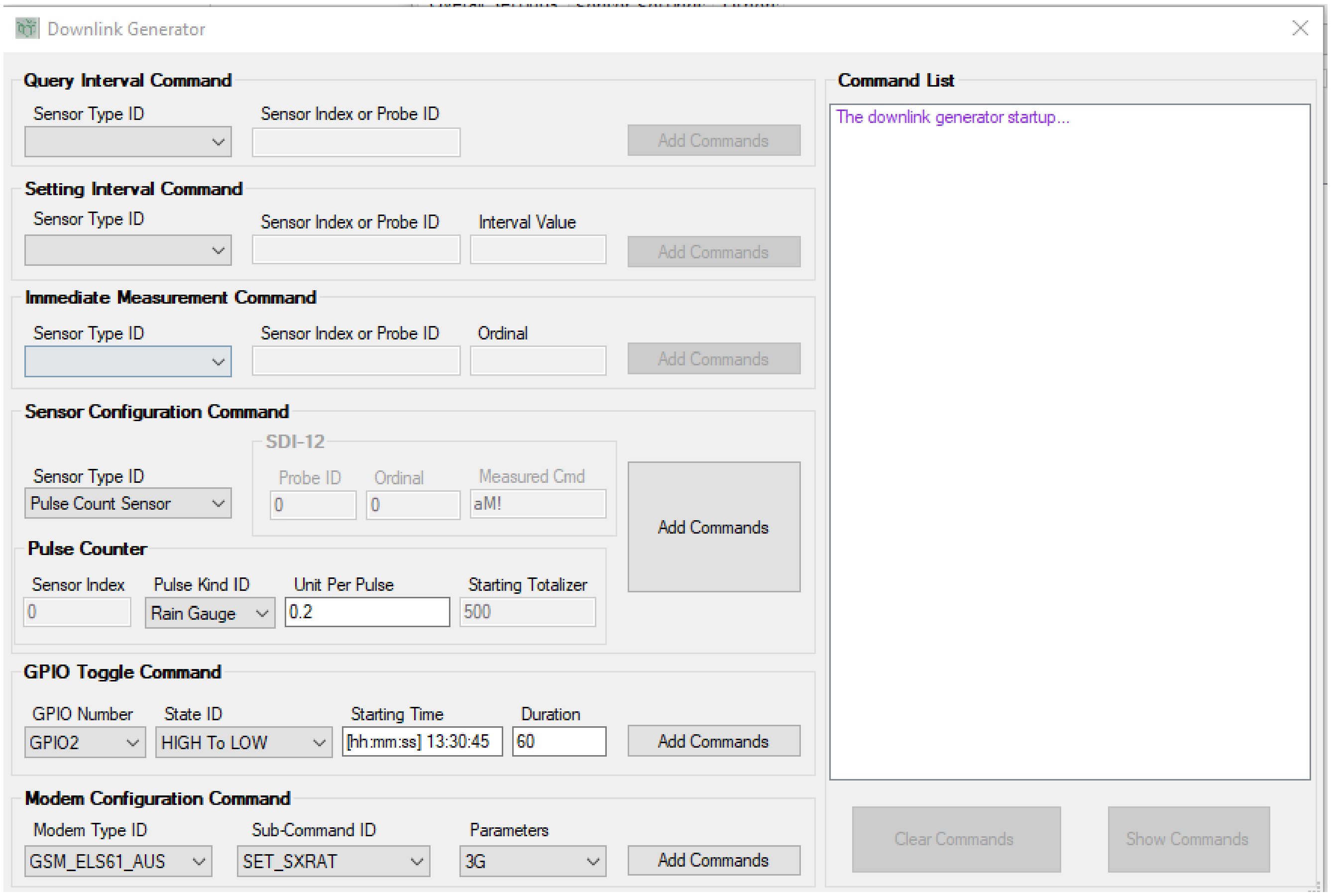
Reserved for Future Use – TBSL1 HW upgrade required.

4.5 Downlink generator

This menu provides a tool that will build the commands to be sent by an application server to TBSL1 to control it remotely (over LoRa or cellular interface).

Available commands allow to:

- Query time interval
  - Sensors measurement interval (SDI-12, pulse, analog)
  - Battery reporting interval
  - Transmission interval
- Set time interval
  - Sensors measurement interval (SDI-12, pulse, analog)
  - Battery reporting interval
  - Transmission interval
- Take an immediate measurement on a sensor
  - SDI-12, analog or pulse sensor
- Update sensors configuration
  - SDI-12: update SDI-12 commands
  - Pulse: update parameters for rain gauge and flow meter applications.
- Trigger GPIO pulses (***Reserved for Future Use – TBSL1 HW upgrade required***)
  - Pulse duration with either high-to-low or low-to-high transitions with starting time alarm.
- Modem configuration
  - LoRaWAN: Class, ADR
  - Cellular: RAT mode and preferred RAT.



Sensor Configuration Command

Sensor Type ID: Pulse Count Sensor

SDI-12: Probe ID: 0; Ordinal: 0; Measured Cmd: aM!

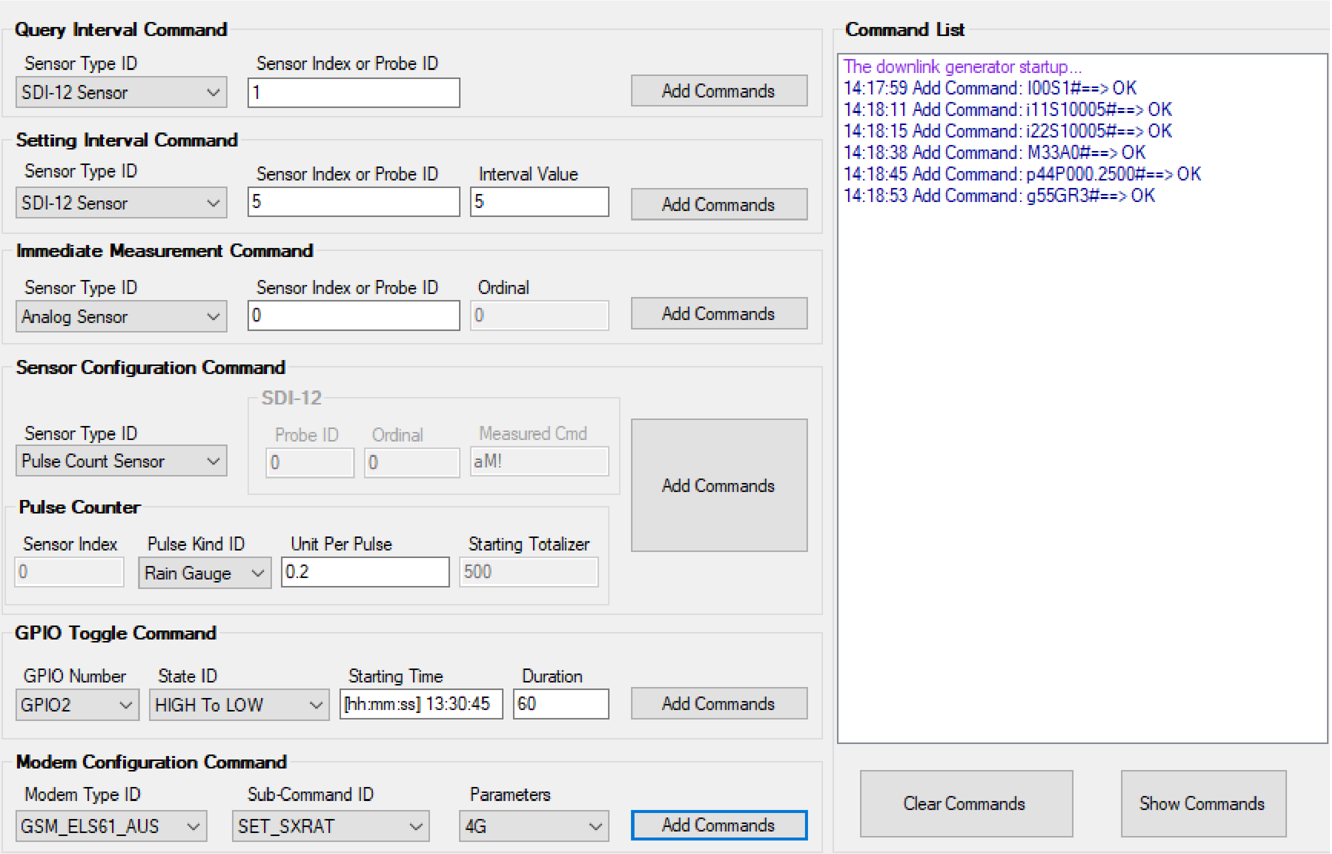
Pulse Counter: Sensor Index: 0; Pulse Kind ID: Rain Gauge; Unit Per Pulse: 0.2; Starting Totalizer: 500

GPIO Toggle Command: GPIO Number: GPIO2; State ID: HIGH To LOW; Starting Time: [hh:mm:ss] 13:30:45; Duration: 60

Modem Configuration Command: Modem Type ID: GSM_ELS61_AUS; Sub-Command ID: SET_SXRAT; Parameters: 3G

Command List: The downlink generator startup...

The user defines and adds the commands he wants to generate by clicking on each ‘Add commands’ buttons whenever applicable. The command list is then populated accordingly.



Query Interval Command

Sensor Type ID: SDI-12 Sensor
Sensor Index or Probe ID: 1

Setting Interval Command

Sensor Type ID: SDI-12 Sensor
Sensor Index or Probe ID: 5
Interval Value: 5

Immediate Measurement Command

Sensor Type ID: Analog Sensor
Sensor Index or Probe ID: 0
Ordinal: 0

Sensor Configuration Command

Sensor Type ID: Pulse Count Sensor

SDI-12
Probe ID: 0
Ordinal: 0
Measured Cmd: aM!

Pulse Counter
Sensor Index: 0
Pulse Kind ID: Rain Gauge
Unit Per Pulse: 0.2
Starting Totalizer: 500

GPIO Toggle Command

GPIO Number: GPIO2
State ID: HIGH To LOW
Starting Time: [hh:mm:ss] 13:30:45
Duration: 60

Modem Configuration Command

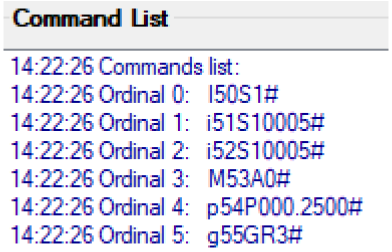
Modem Type ID: GSM_ELS61_AUS
Sub-Command ID: SET_SXRAT
Parameters: 4G

Command List

```
The downlink generator startup...
14:17:59 Add Command: I00S1#==> OK
14:18:11 Add Command: i11S10005#==> OK
14:18:15 Add Command: i22S10005#==> OK
14:18:38 Add Command: M33A0#==> OK
14:18:45 Add Command: p44P000.2500#==> OK
14:18:53 Add Command: g55GR3#==> OK
```

For each command category, multiple commands can be generated (e.g it's possible to set the measurement interval for five SDI-12 sensors and one analog sensor).

Then click on 'Show Commands' to display the list of commands that can be sent by the application server to control the platform:



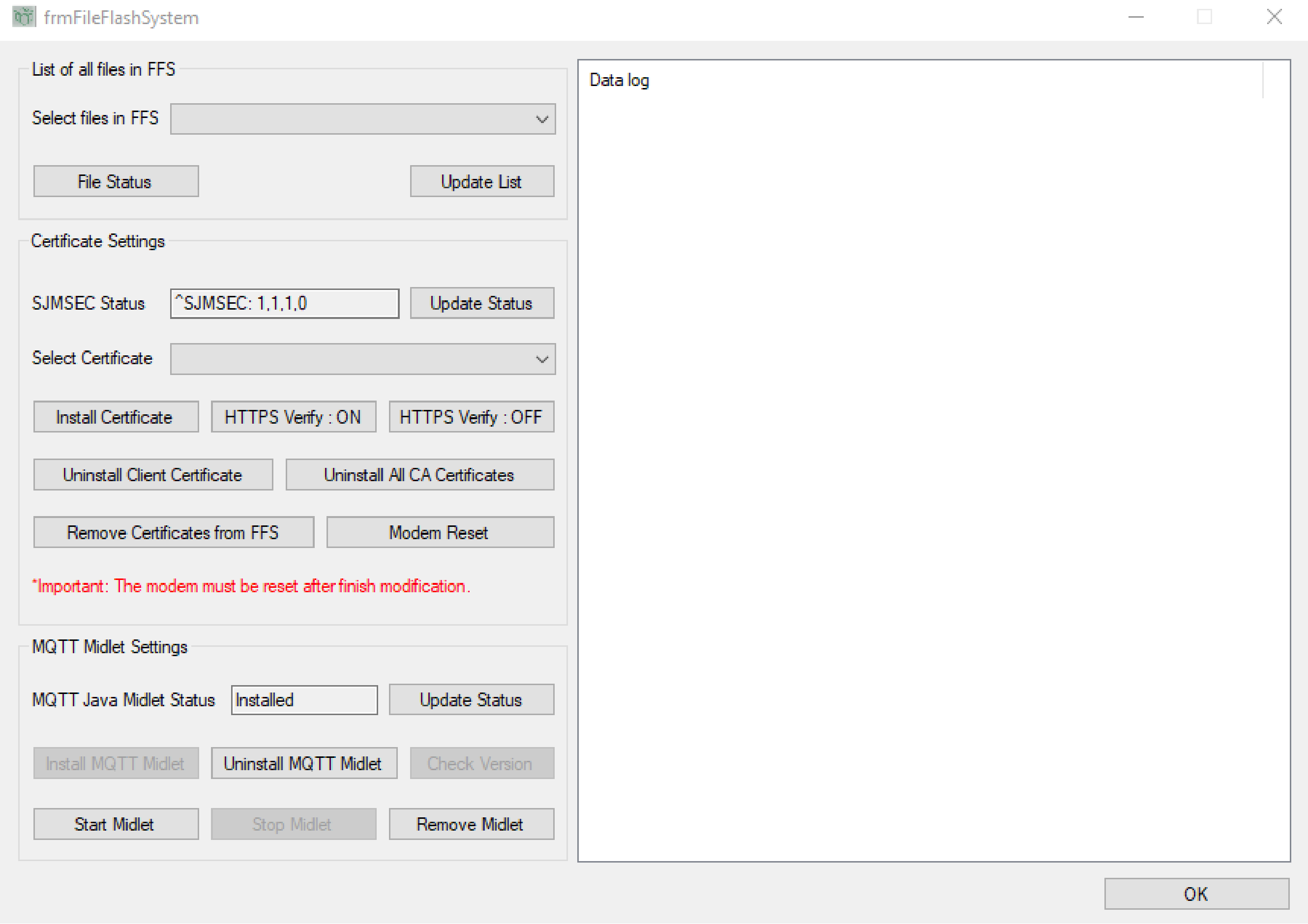
Command List

```
14:22:26 Commands list:
14:22:26 Ordinal 0:   I50S1#
14:22:26 Ordinal 1:   i51S10005#
14:22:26 Ordinal 2:   i52S10005#
14:22:26 Ordinal 3:   M53A0#
14:22:26 Ordinal 4:   p54P000.2500#
14:22:26 Ordinal 5:   g55GR3#
```

4.6 Flash File System

This configuration panel is available only for cellular modems.

It is used to access the flash file system of the cellular modem and handle MQTT certificates and application MIDlet.



SJMSEC Status: ^SJMSEC: 1,1,1,0

*Important: The modem must be reset after finish modification.

MQTT Java Midlet Status: Installed

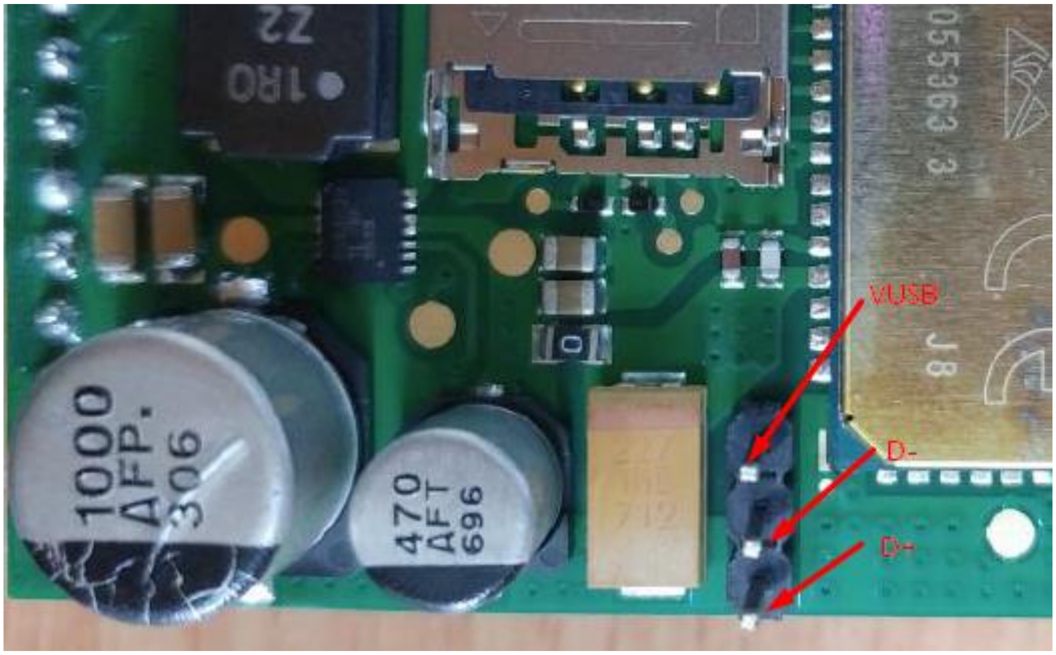
It is then possible through this menu to:

- List files in FFS
- Install/Uninstall/Start/Stop/Remove MIDlet
- Install/Uninstall/Remove TLS1.2 client certificate
- Install/Uninstall/Remove TLS1.2 CA certificate(s) (**[NOTE]: the modem does not support CA bundle, such bundle must be split into individual CA files, loaded and installed one by one).**

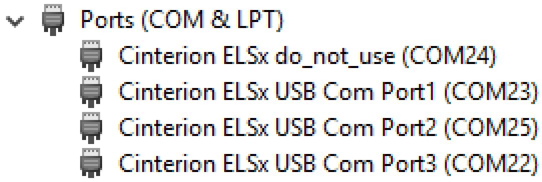
4.6.1 Prerequisites

The modem driver needs to be installed on the PC prior to any configuration change. Contact Tekbox technical support to get the latest version of the driver.

Once installed, connect the 3 pins connector to a PC USB 2.0 port:



From the device manager, several USB COM port related to the modem ('Cinterion') should appear:

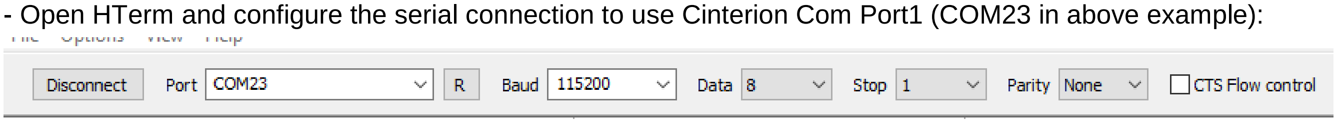


Ports (COM & LPT)
- Cinterion ELSx do_not_use (COM24)
- Cinterion ELSx USB Com Port1 (COM23)
- Cinterion ELSx USB Com Port2 (COM25)
- Cinterion ELSx USB Com Port3 (COM22)

4.6.2 Copy files to modem FFS

- Install HTerm terminal application on the PC. It can be downloaded from http://www.der-hammer.info/terminal/

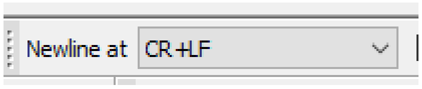
- Open HTerm and configure the serial connection to use Cinterion Com Port1 (COM23 in above example):



Disconnect | Port COM23 | R | Baud 115200 | Data 8 | Stop 1 | Parity None | □ CTS Flow control

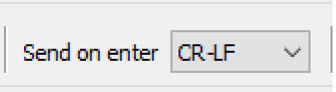
If this does not work, use the next available port.

- Configure CR+LF for sending and receiving data:



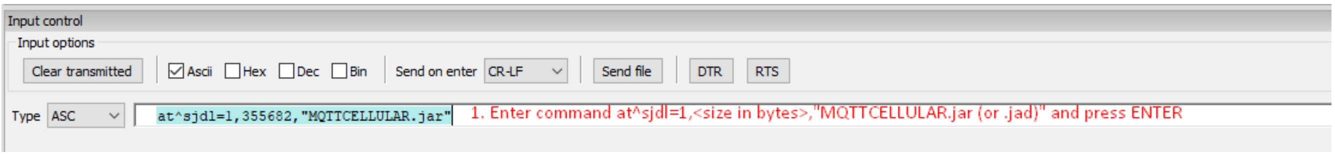
Newline at CR+LF

and



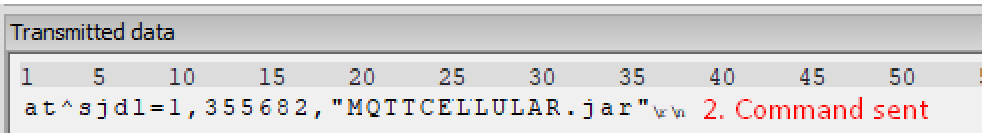
Send on enter CR-LF

- Copy the MIDlet to FFS (both .jad and .jar files):



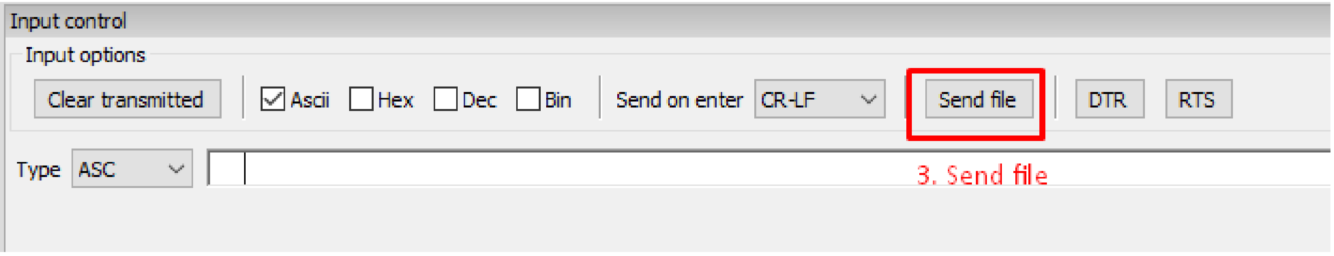
```
at^sjdl=1,355682,"MQTTCELLULAR.jar"
```

1. Enter command at^sjdl=1,<size in bytes>,"MQTTCELLULAR.jar (or .jad)" and press ENTER

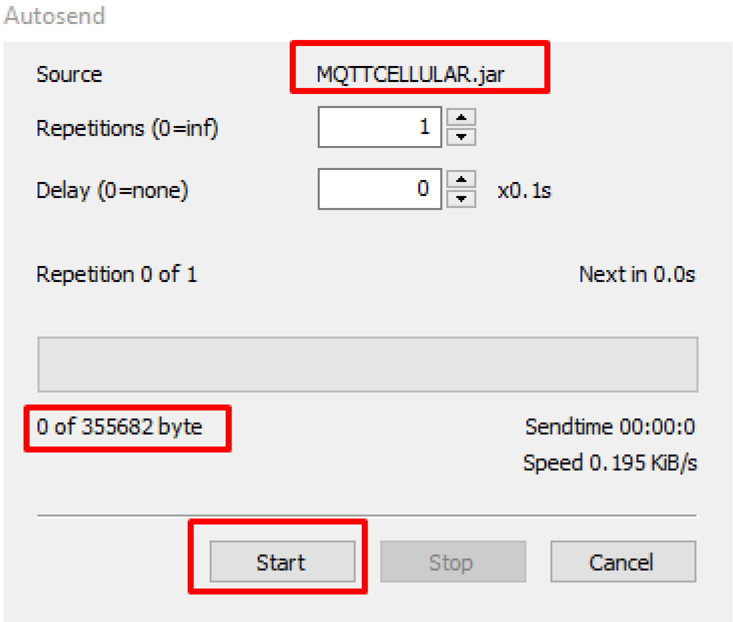


```
at^sjdl=1,355682,"MQTTCELLULAR.jar"\r\n
```

2. Command sent



3. Send file



Source MQTTCELLULAR.jar

Repetitions (0=inf) 1

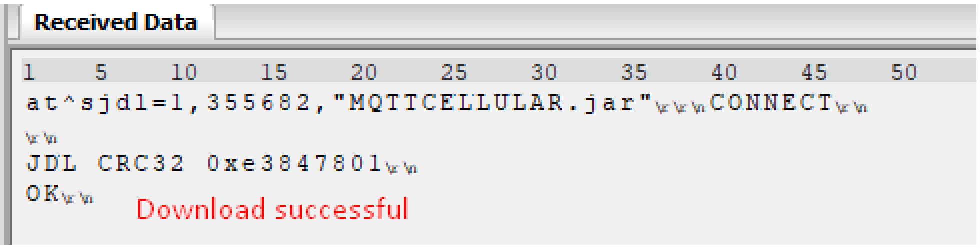
Delay (0=none) 0 x0.1s

Repetition 0 of 1 Next in 0.0s

0 of 355682 byte

Sendtime 00:00:0

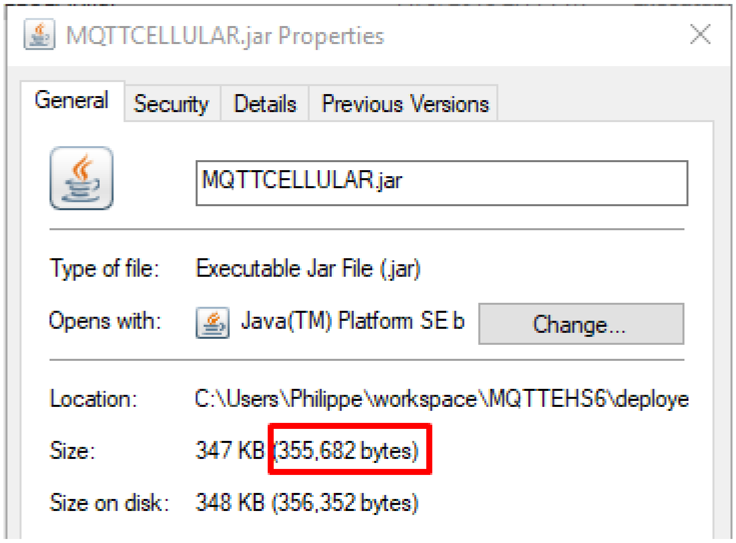
Speed 0.195 KiB/s



```
at^sjdl=1,355682,"MQTTCELLULAR.jar"\r\r\nCONNECT\r\n
\r\n
JDL CRC32 0xe3847801\r\n
OK\r\n
```

Download successful

File size can be checked by right clicking the file name from Windows explorer and checking the properties.



MQTTCELLULAR.jar

Type of file: Executable Jar File (.jar)

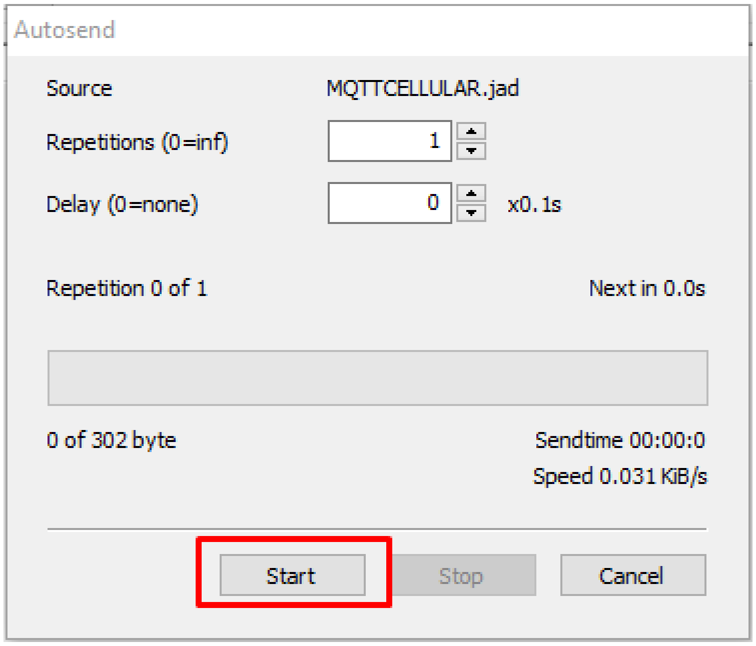
Opens with: Java(TM) Platform SE b

Location: C:\Users\Philippe\workspace\MQTTEHS6\deploye

Size: 347 KB (355,682 bytes)

Size on disk: 348 KB (356,352 bytes)

Then click on ‘Send File’, select the file and send it.



Source MQTTCELLULAR.jad

Repetitions (0=inf) 1

Delay (0=none) 0 x0.1s

Repetition 0 of 1

Next in 0.0s

0 of 302 byte

Sendtime 00:00:0

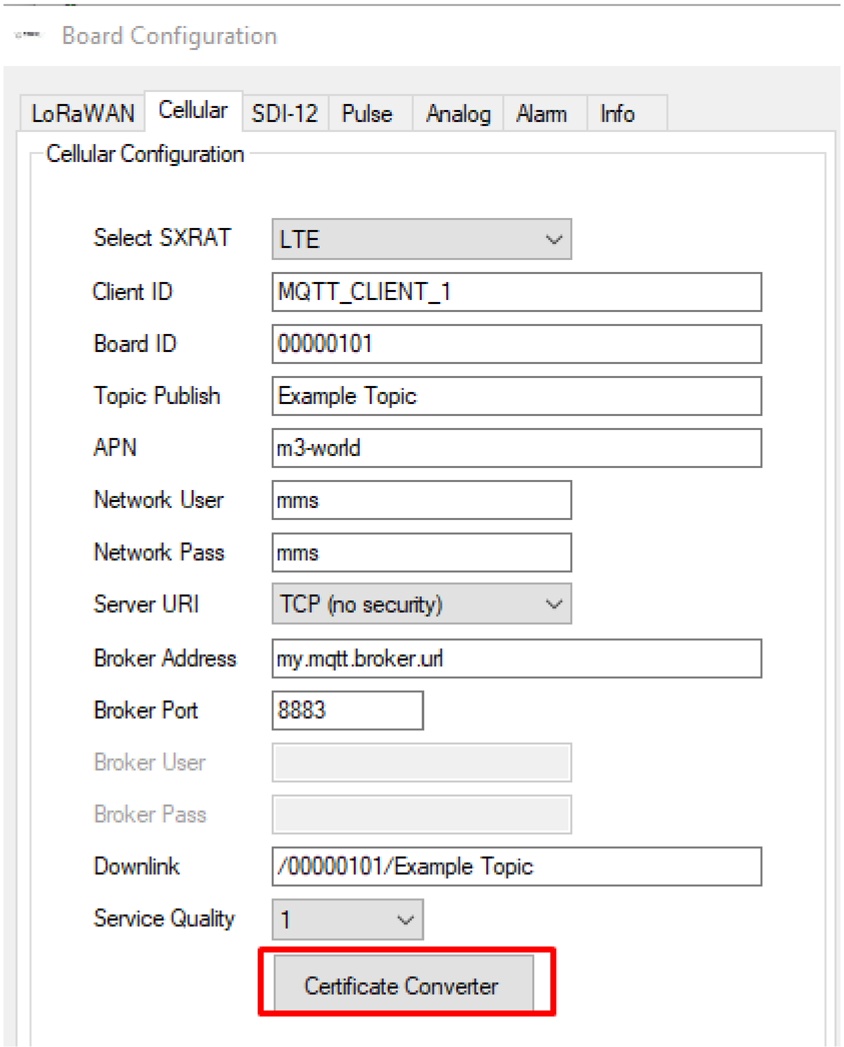
Speed 0.031 KiB/s

- Copy CA certificate(s) and client certificate/key (if needed): steps are the same than for downloading the MIDlet but there are some preliminary steps required as only certificates in binary format can be used.

Certificates and ciphering key need to comply with the format and extensions constraints:

- CA certificate(s) must be in DER format and have .der extension
- Client certificate must be in PEM format and have .crt extension and its ciphering key must be in PEM format and have .key extension

Certificates and ciphering key need then to be converted to binary format before being installed. This is achieved by using the converter embedded within the configuration tool.



Board Configuration

Cellular Configuration

Select SXRAT: LTE

Client ID: MQTT_CLIENT_1

Board ID: 00000101

Topic Publish: Example Topic

APN: m3-world

Network User: mms

Network Pass: mms

Server URI: TCP (no security)

Broker Address: my.mqtt.broker.url

Broker Port: 8883

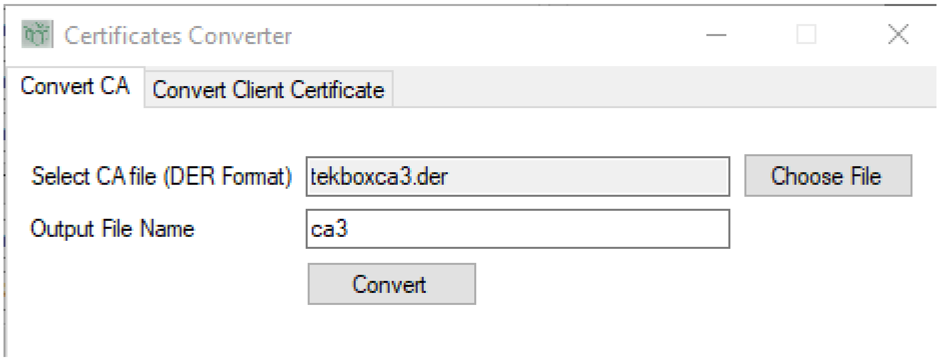
Broker User:

Broker Pass:

Downlink: /00000101/Example Topic

Service Quality: 1

Choose the CA certificate (or client certificate and its key) then enter the desired output file name (no extension required as it will automatically be saved as .bin file).



Certificates Converter

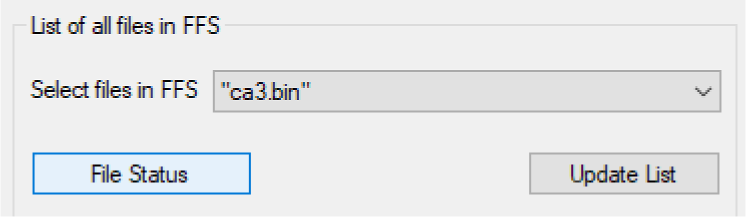
Select CA file (DER Format): tekboxca3.der

Output File Name: ca3

The generated <certificate>.bin file is then ready to be stored into the modem FFS.

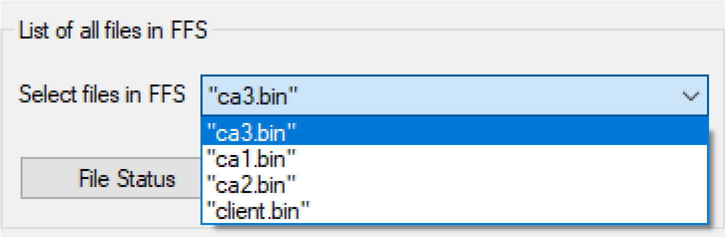
4.6.3 List of all files in FFS

- Click on ‘Update List’ to get the content of FFS in the drop down menu



List of all files in FFS

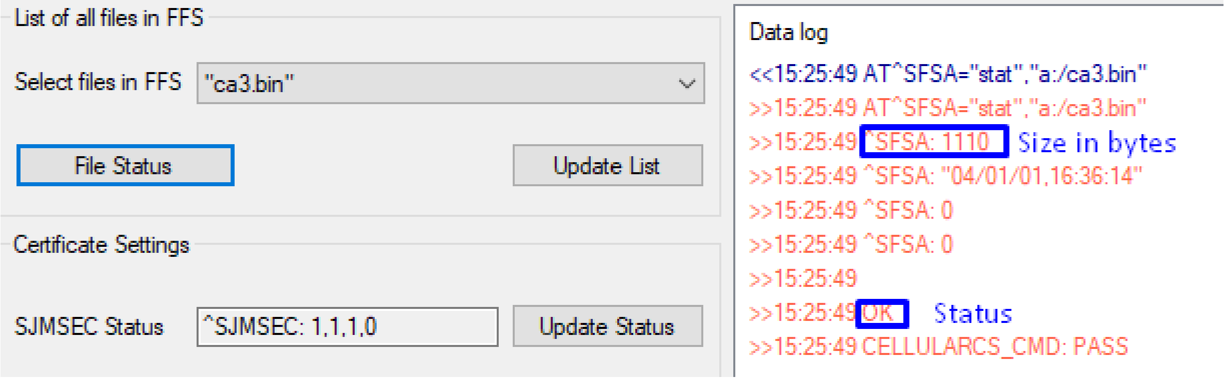
Select files in FFS: "ca3.bin"



List of all files in FFS

Select files in FFS: "ca3.bin"

- Click on ‘File Status’ to get information about a particular file from FFS



List of all files in FFS

Select files in FFS: "ca3.bin"

Certificate Settings

SJMSEC Status: ^SJMSEC: 1,1,1,0

```
<<15:25:49 AT^SFSA="stat","a:/ca3.bin"
>>15:25:49 AT^SFSA="stat","a:/ca3.bin"
>>15:25:49 ^SFSA: 1110
>>15:25:49 ^SFSA: "04/01/01,16:36:14"
>>15:25:49 ^SFSA: 0
>>15:25:49 ^SFSA: 0
>>15:25:49
>>15:25:49 OK
>>15:25:49 CELLULARCS_CMD: PASS
```

4.6.4 Certificates settings

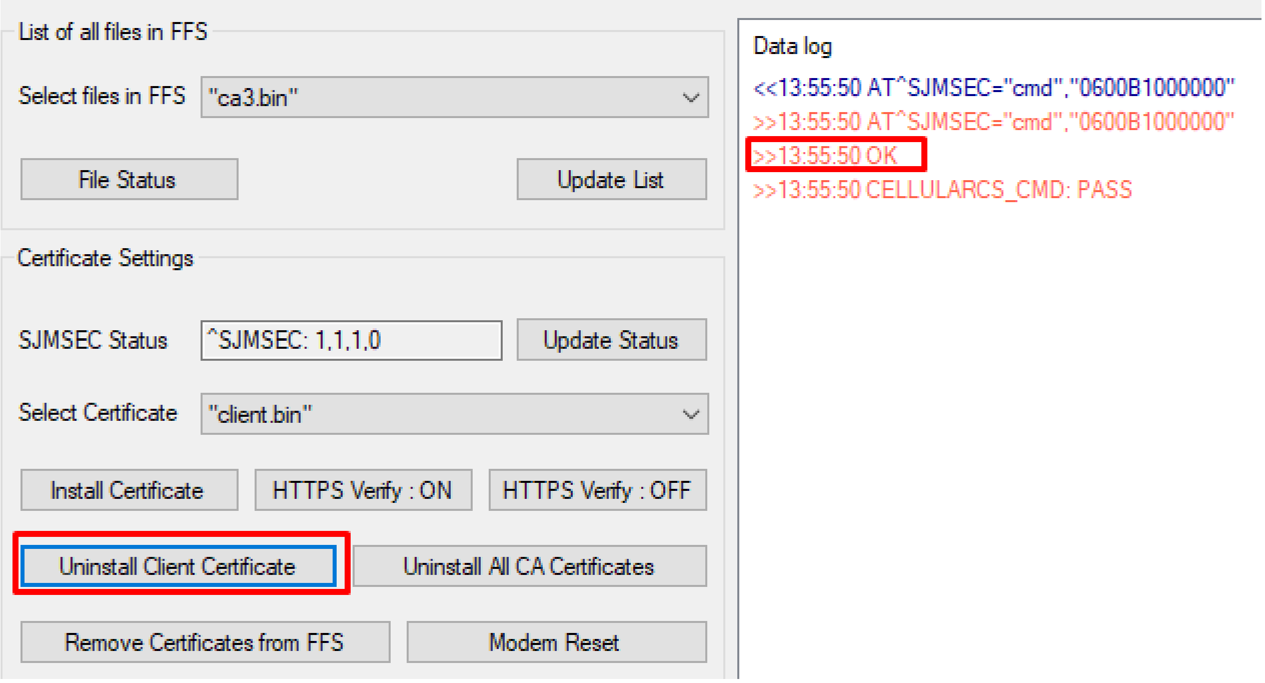
Certificates must have been downloaded to FFS prior to any configuration action.

A certificate must be installed so it becomes active in the modem.
A certificate must be uninstalled before being physically removed from FFS.

Only one client certificate can be installed, whereas multiple CA can be installed (this is the case when CA bundle is used).

Certificates must have been converted from DER to BIN format through the conversion tool before being installed in the modem (cf related section for further details).

- Uninstall client certificate



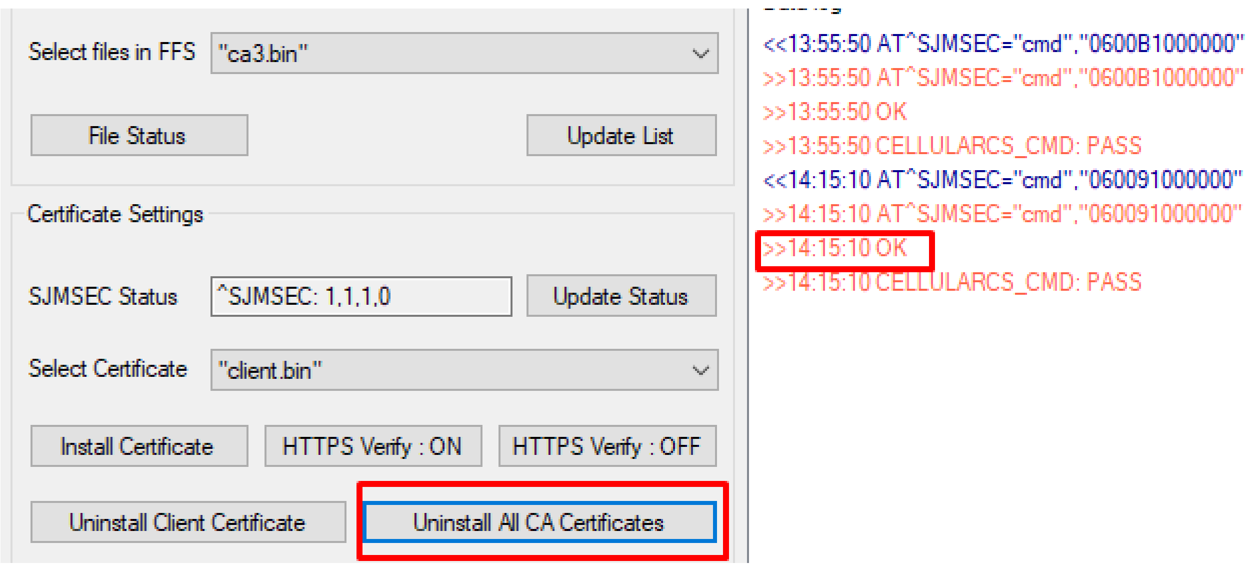
Select files in FFS: "ca3.bin"

SJMSEC Status: ^SJMSEC: 1,1,1,0

Select Certificate: "client.bin"

```
<<13:55:50 AT^SJMSEC="cmd","0600B1000000"
>>13:55:50 AT^SJMSEC="cmd","0600B1000000"
>>13:55:50 OK
>>13:55:50 CELLULARCS_CMD: PASS
```

- Uninstall all CA certificates



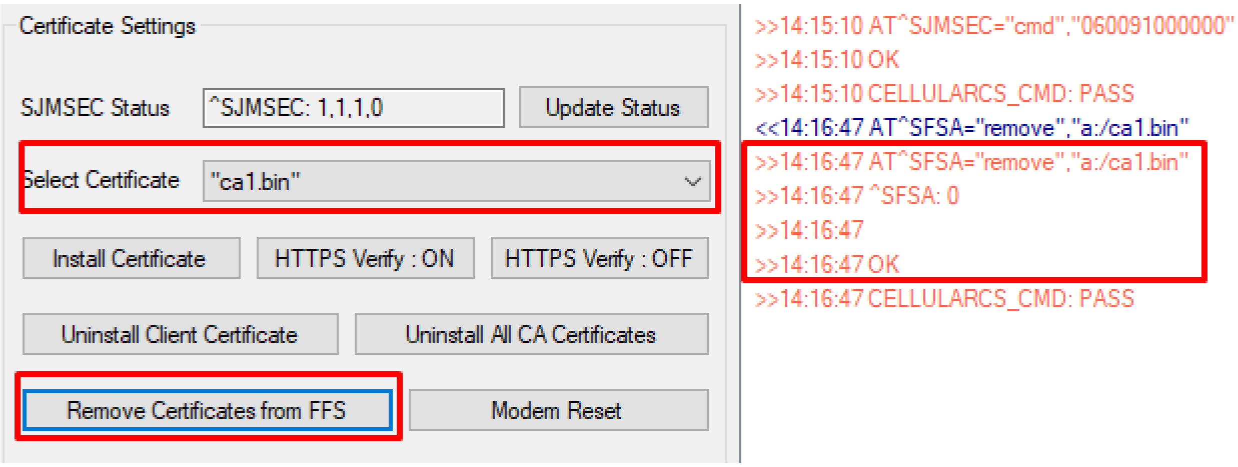
Select files in FFS: "ca3.bin"

SJMSEC Status: ^SJMSEC: 1,1,1,0

Select Certificate: "client.bin"

```
<<13:55:50 AT^SJMSEC="cmd","0600B1000000"
>>13:55:50 AT^SJMSEC="cmd","0600B1000000"
>>13:55:50 OK
>>13:55:50 CELLULARCS_CMD: PASS
<<14:15:10 AT^SJMSEC="cmd","060091000000"
>>14:15:10 AT^SJMSEC="cmd","060091000000"
>>14:15:10 OK
>>14:15:10 CELLULARCS_CMD: PASS
```

- Remove a certificate from FFS

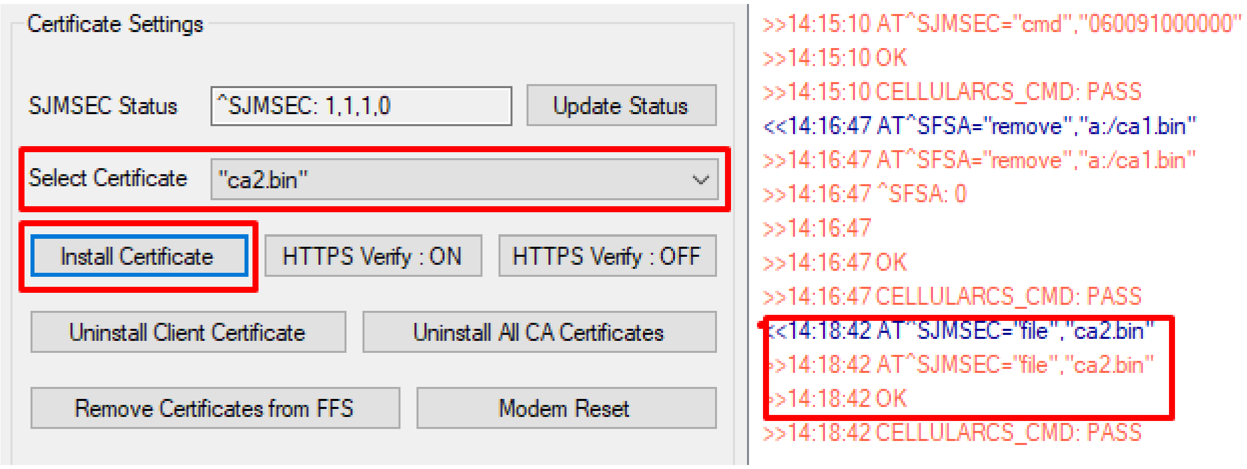


SJMSEC Status: ^SJMSEC: 1,1,1,0

Select Certificate: "ca1.bin"

```
>>14:15:10 AT^SJMSEC="cmd","060091000000"
>>14:15:10 OK
>>14:15:10 CELLULARCS_CMD: PASS
<<14:16:47 AT^SFSA="remove","a:/ca1.bin"
>>14:16:47 AT^SFSA="remove","a:/ca1.bin"
>>14:16:47 ^SFSA: 0
>>14:16:47
>>14:16:47 OK
>>14:16:47 CELLULARCS_CMD: PASS
```

- Install a certificate

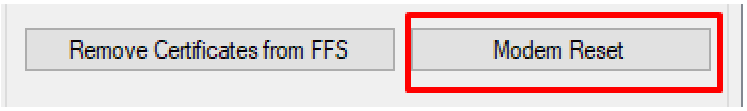


SJMSEC Status: ^SJMSEC: 1,1,1,0

Select Certificate: "ca2.bin"

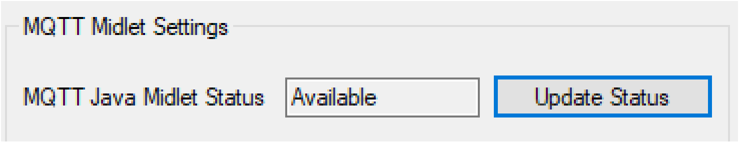
```
>>14:15:10 AT^SJMSEC="cmd","060091000000"
>>14:15:10 OK
>>14:15:10 CELLULARCS_CMD: PASS
<<14:16:47 AT^SFSA="remove","a:/ca1.bin"
>>14:16:47 AT^SFSA="remove","a:/ca1.bin"
>>14:16:47 ^SFSA: 0
>>14:16:47
>>14:16:47 OK
>>14:16:47 CELLULARCS_CMD: PASS
<<14:18:42 AT^SJMSEC="file","ca2.bin"
>>14:18:42 AT^SJMSEC="file","ca2.bin"
>>14:18:42 OK
>>14:18:42 CELLULARCS_CMD: PASS
```

- Reset modem once all operations have been completed



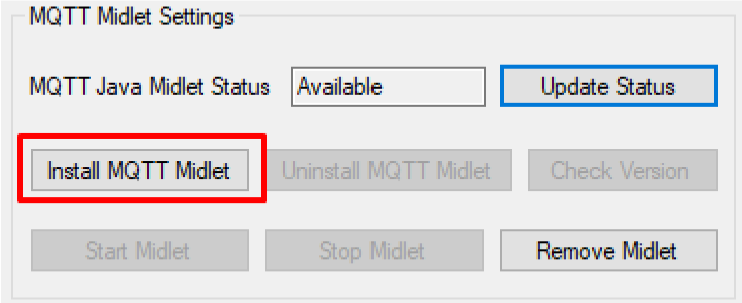
4.6.5 MQTT MIDlet settings

- Once downloaded to the modem FFS the MIDlet appears as “Available”:

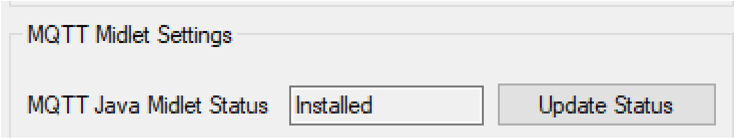


MQTT Java Midlet Status: Available

- Install the MIDlet, its status will switch to “Installed”

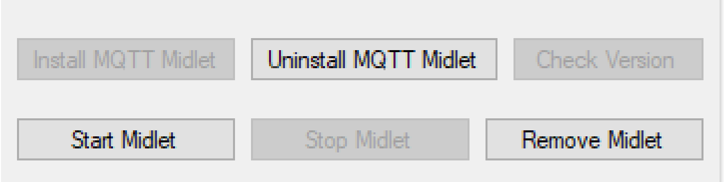


MQTT Java Midlet Status: Available

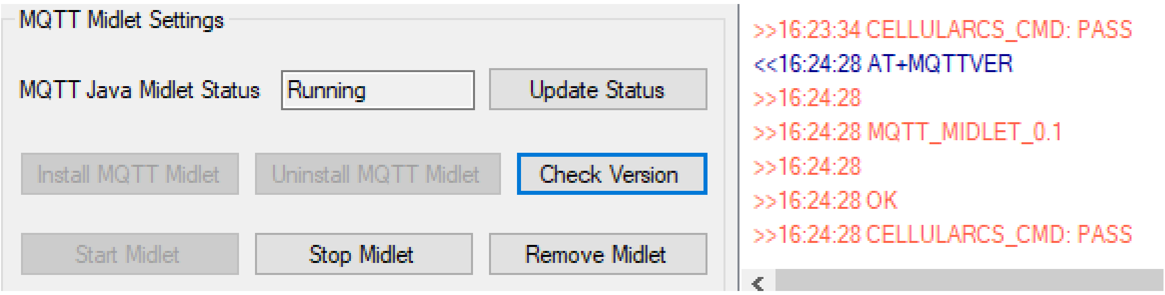


MQTT Java Midlet Status: Installed

- The MIDlet can then be started (required to leverage MQTT feature), uninstalled or removed from FFS.



- Once running, the MIDlet version can be checked:



MQTT Java Midlet Status: Running

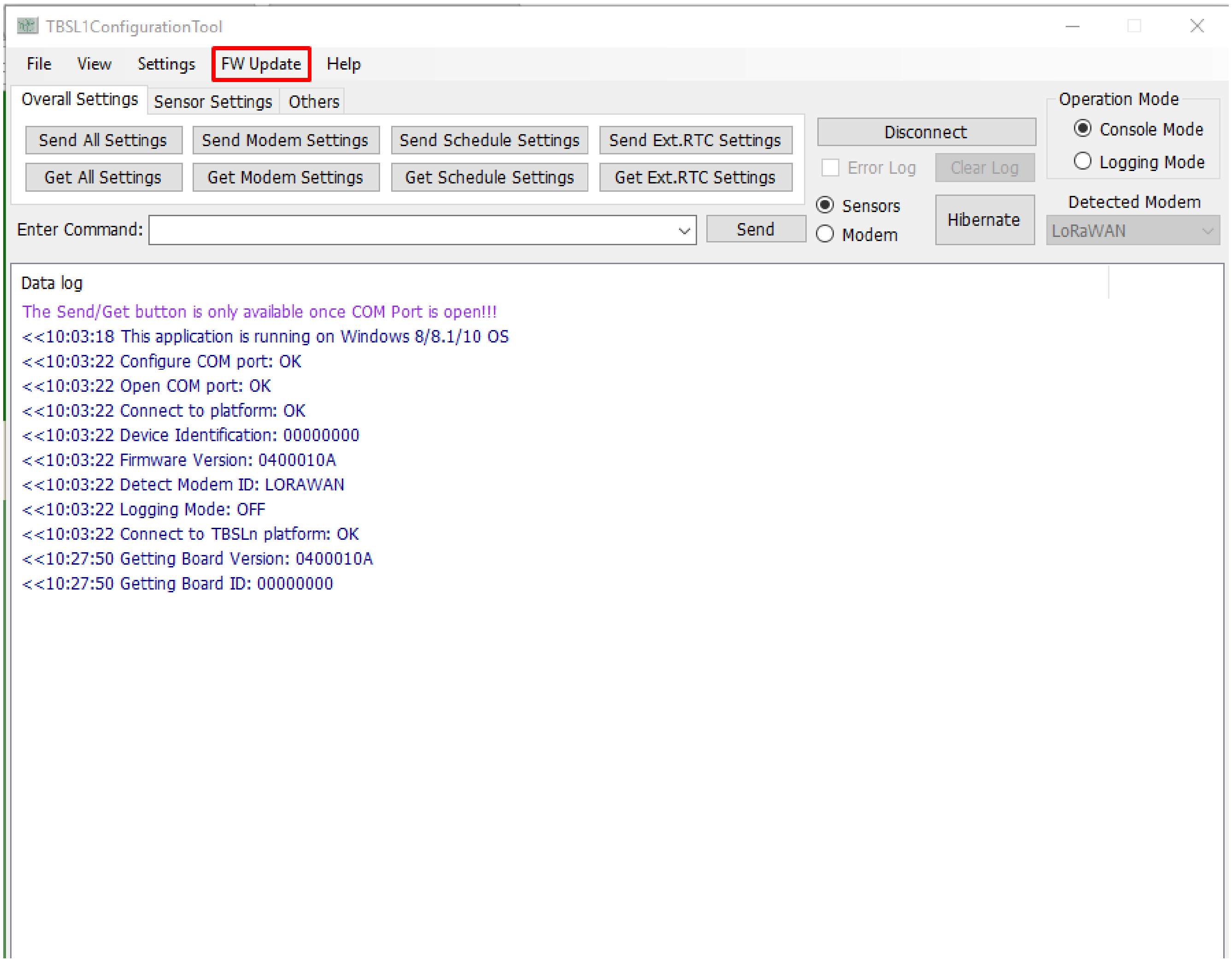
```
>>16:23:34 CELLULARCS_CMD: PASS
<<16:24:28 AT+MQTTVER
>>16:24:28
>>16:24:28 MQTT_MIDLET_0.1
>>16:24:28
>>16:24:28 OK
>>16:24:28 CELLULARCS_CMD: PASS
```

- When the MIDlet is removed from FFS it does not uninstall it, these are two different things.
To uninstall the MIDlet it must be first stopped (if it’s running) and only then it can be uninstalled.

- Finally reset the modem by clicking on ‘Modem Reset’ and once completed reset the platform.

4.7 FW Update Menu

TBSL1 configuration tool embeds a firmware update feature via USB cable.



```
The Send/Get button is only available once COM Port is open!!!
<<10:03:18 This application is running on Windows 8/8.1/10 OS
<<10:03:22 Configure COM port: OK
<<10:03:22 Open COM port: OK
<<10:03:22 Connect to platform: OK
<<10:03:22 Device Identification: 00000000
<<10:03:22 Firmware Version: 0400010A
<<10:03:22 Detect Modem ID: LORAWAN
<<10:03:22 Logging Mode: OFF
<<10:03:22 Connect to TBSLn platform: OK
<<10:27:50 Getting Board Version: 0400010A
<<10:27:50 Getting Board ID: 00000000
```

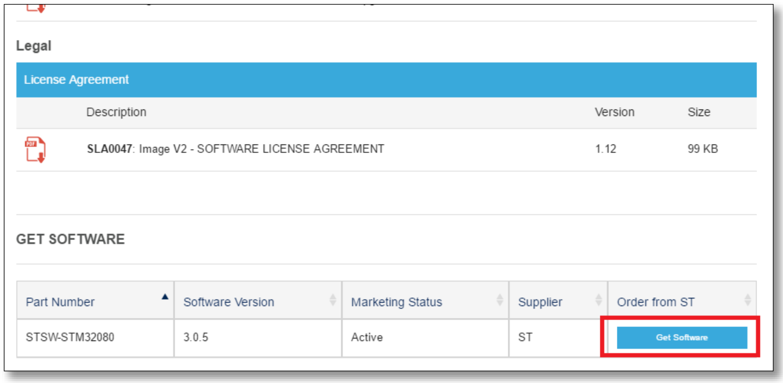
Fig.17 FW Update Menu.

Hardware requirement:

1. USB cable connection between the platform and PC.

Software requirement:

1. Visit website: *http://www.st.com/en/development-tools/stsw-stm32080.html*
2. Download DfuSe tool and install it.



Legal

License Agreement

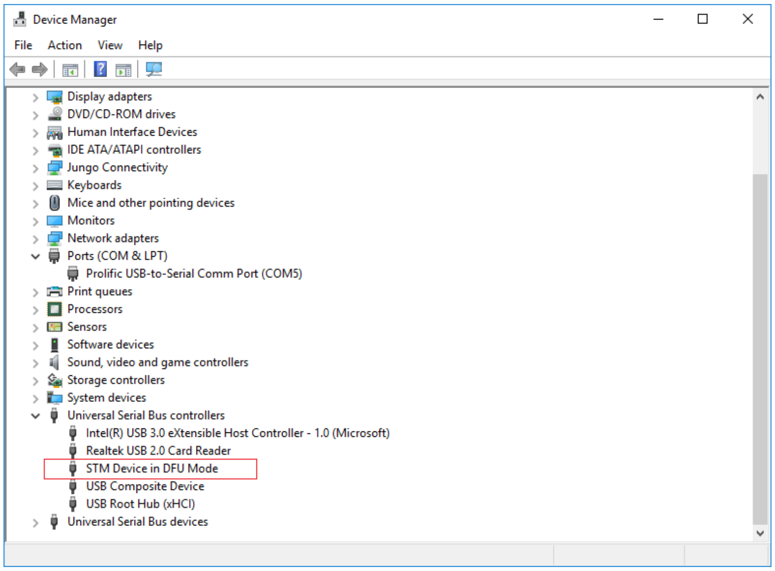
	Description	Version	Size
	SLA0047: Image V2 - SOFTWARE LICENSE AGREEMENT	1.12	99 KB

GET SOFTWARE

Part Number	Software Version	Marketing Status	Supplier	Order from ST
STSW-STM32080	3.0.5	Active	ST	Get Software

Fig.18 Download DFUse Tool from ST website.

3. Driver for STM Device in DFU Mode.



- Display adapters
- DVD/CD-ROM drives
- Human Interface Devices
- IDE ATA/ATAPI controllers
- Jungo Connectivity
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Ports (COM & LPT)
  - Prolific USB-to-Serial Comm Port (COM5)
- Print queues
- Processors
- Sensors
- Software devices
- Sound, video and game controllers
- Storage controllers
- System devices
- Universal Serial Bus controllers
  - Intel(R) USB 3.0 eXtensible Host Controller - 1.0 (Microsoft)
  - Realtek USB 2.0 Card Reader
  - STM Device in DFU Mode
  - USB Composite Device
  - USB Root Hub (xHCI)
- Universal Serial Bus devices

Fig.19 Driver for STM Device in DFU Mode.

Note: Normally, this driver will be automatically installed when it is detected by Windows (tested with Window10 x64). If the PC doesn’t automatically install the driver, this can be done manually by following these steps:

- Go to installation folder of DfuSe Tool:
  Installed Disk\STMicroelectronics\Software\DfuSe v3.0.5\Bin\Driver
- Select folder compatible with your PC window version. (Window 7, 8, 8.1)
- Select folder compatible with your PC. (x64, x86)
- Run dpinst_amd64.exe or dpinst_amd86.exe file.

How to use:

1. Open the GUI in console mode.
2. Click on Update item in FW Update menu.

3. Wait for the platform to switch to DFU Bootloader mode.

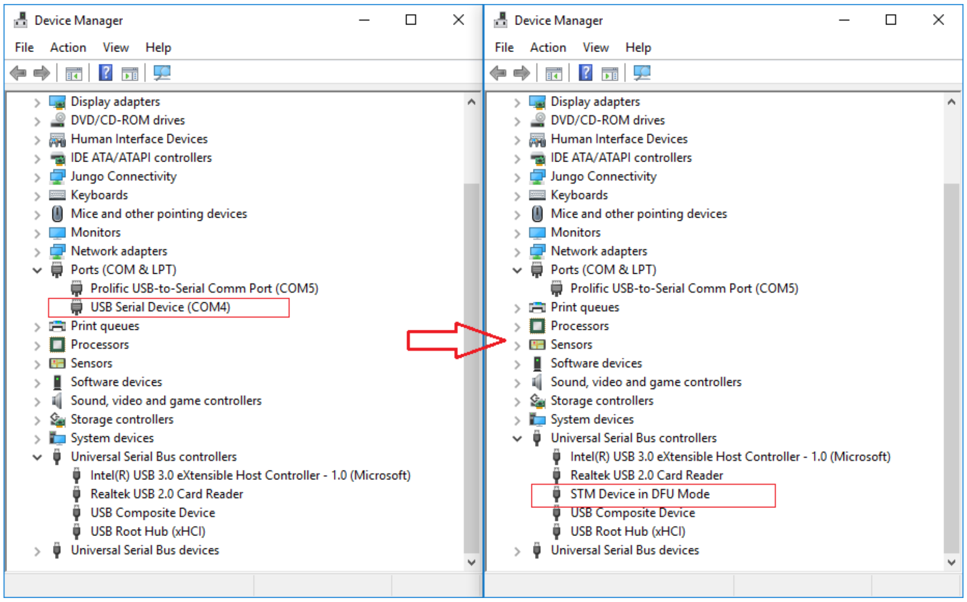
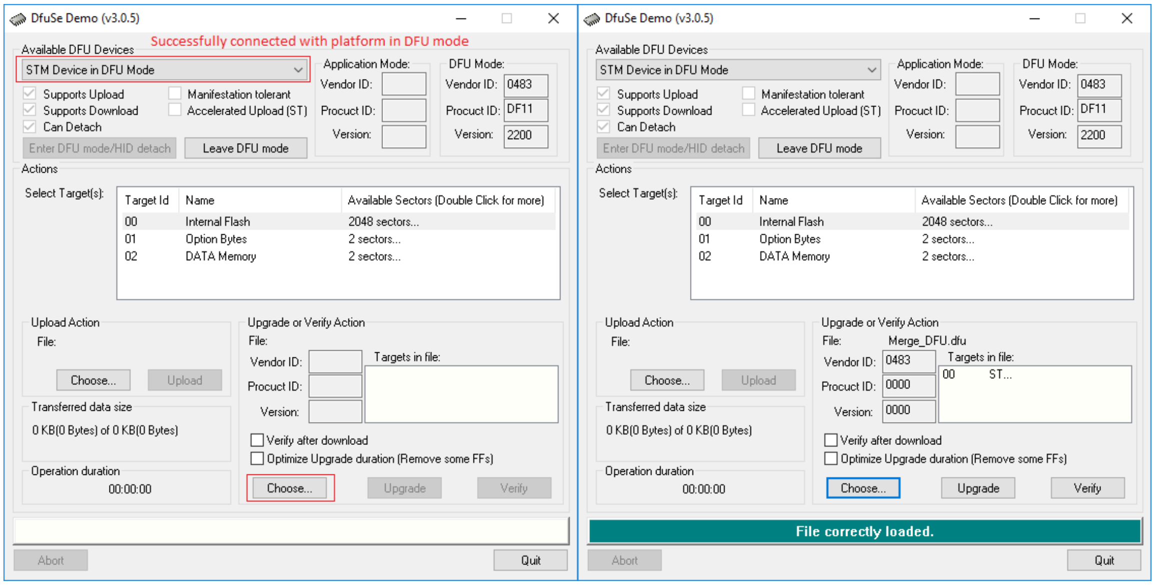


Fig.20 Platform switched to DFU Bootloader mode.

4. Open DFUse Tool and click ***Choose*** to select provided dfu file (DFU file *.dfu).



Successfully connected with platform in DFU mode

STM Device in DFU Mode

Vendor ID:	0483
Procuct ID:	DF11
Version:	2200

Target Id	Name	Available Sectors (Double Click for more)
00	Internal Flash	2048 sectors...
01	Option Bytes	2 sectors...
02	DATA Memory	2 sectors...

0 KB(0 Bytes) of 0 KB(0 Bytes)

00:00:00

File: Merge_DFU.dfu

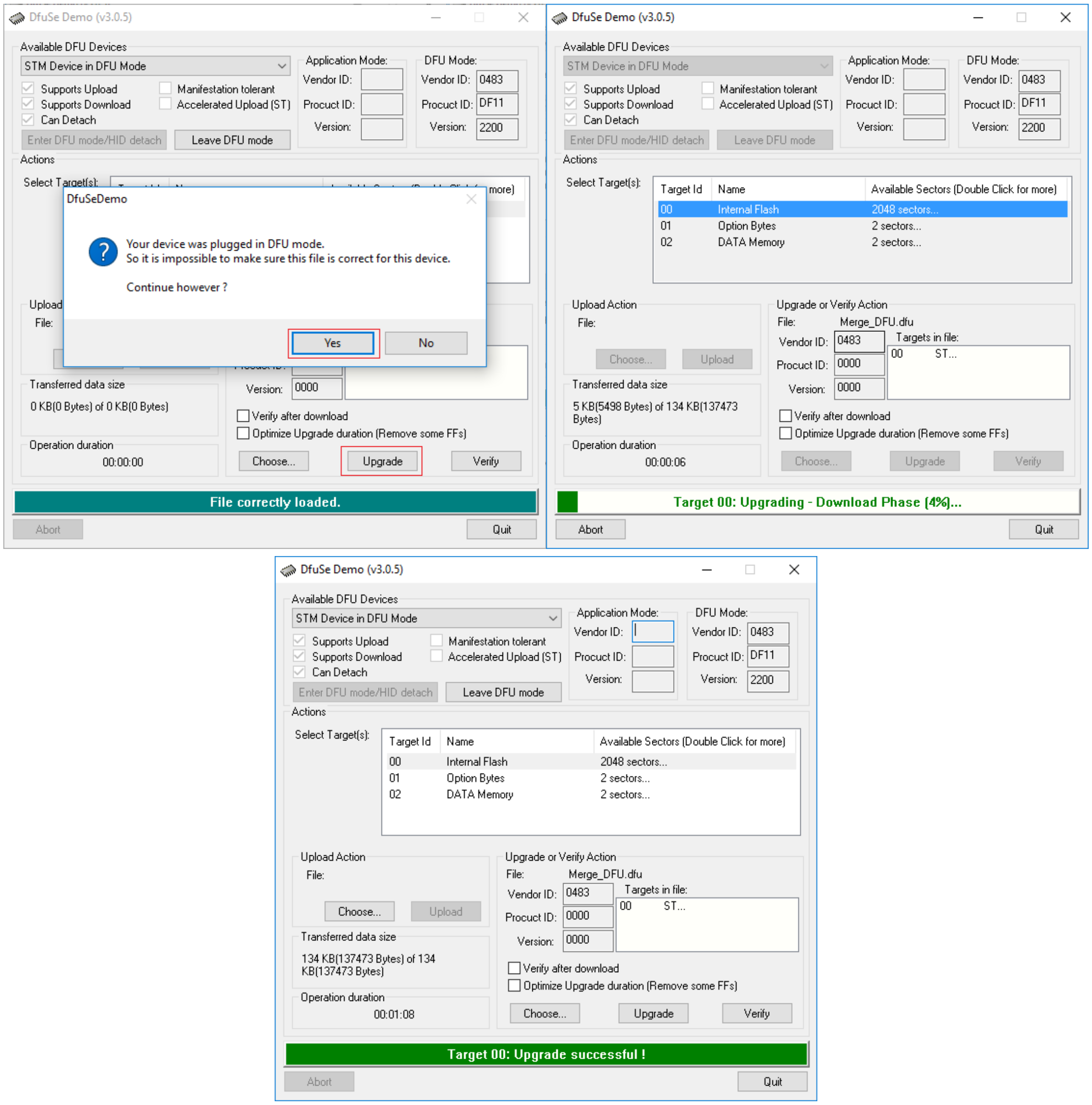
Vendor ID:	0483
Procuct ID:	0000
Version:	0000

Targets in file: 00 ST...

File correctly loaded.

Fig.21 DfuSe Tool.

5. Click ***Upgrade*** to update new firmware, click Yes to close the pop-up message.



DfuSe Demo (v3.0.5)

Available DFU Devices: STM Device in DFU Mode

☑ Supports Upload ☐ Manifestation tolerant
☑ Supports Download ☐ Accelerated Upload (ST)
☑ Can Detach

DFU Mode: Vendor ID: 0483 Procuct ID: DF11 Version: 2200

DfuSeDemo

Your device was plugged in DFU mode.
So it is impossible to make sure this file is correct for this device.

Continue however ?

Yes No

Transferred data size: 0 KB(0 Bytes) of 0 KB(0 Bytes)

Operation duration: 00:00:00

Version: 0000

☐ Verify after download
☐ Optimize Upgrade duration (Remove some FFs)

File correctly loaded.

DfuSe Demo (v3.0.5)

Target Id	Name	Available Sectors (Double Click for more)
00	Internal Flash	2048 sectors...
01	Option Bytes	2 sectors...
02	DATA Memory	2 sectors...

Upgrade or Verify Action — File: Merge_DFU.dfu; Vendor ID: 0483; Procuct ID: 0000; Version: 0000; Targets in file: 00 ST...

Transferred data size: 5 KB(5498 Bytes) of 134 KB(137473 Bytes)

Operation duration: 00:00:06

Target 00: Upgrading - Download Phase (4%)...

DfuSe Demo (v3.0.5)

Transferred data size: 134 KB(137473 Bytes) of 134 KB(137473 Bytes)

Operation duration: 00:01:08

Target 00: Upgrade successful !

Fig.22 Successfully upgrade firmware.

6. Unplug USB cable to exit DFU Bootloader mode and power off TBSL1 by disconnecting the solar panel.
7. Connect back the solar panel and USB cable for further configuration.

4.8 Help Menu.

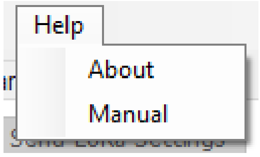
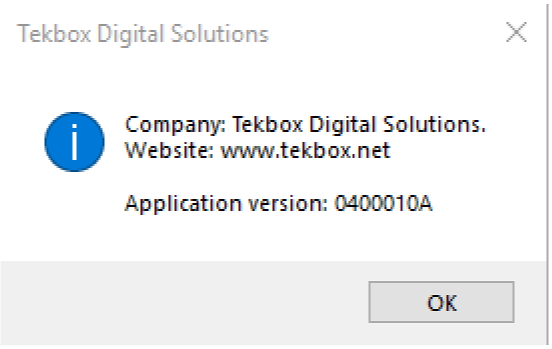


Fig.23 Help Menu.

4.8.1 About

Click on this, TBSL1ConfigurationTool will open a message pop-up with information about company, website, FW version, and the application version.



Tekbox Digital Solutions

Company: Tekbox Digital Solutions.
Website: www.tekbox.net

Application version: 0400010A

Fig.24 TBSL1 Detailed Information.

The application version must match the TBSL1 FW version, otherwise it could result in unpredictable behavior of the platform.

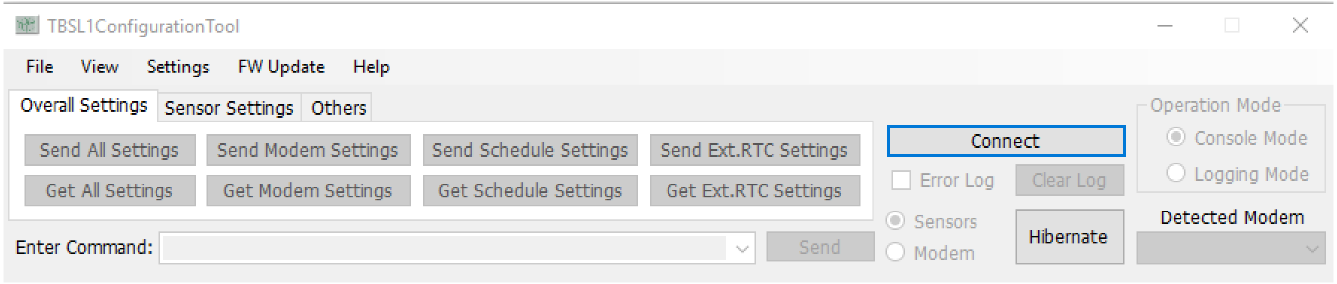
The configuration tool automatically detects if there’s a mismatch between the tool version and the TBSL1 FW version, and notifies the user with a pop up message:

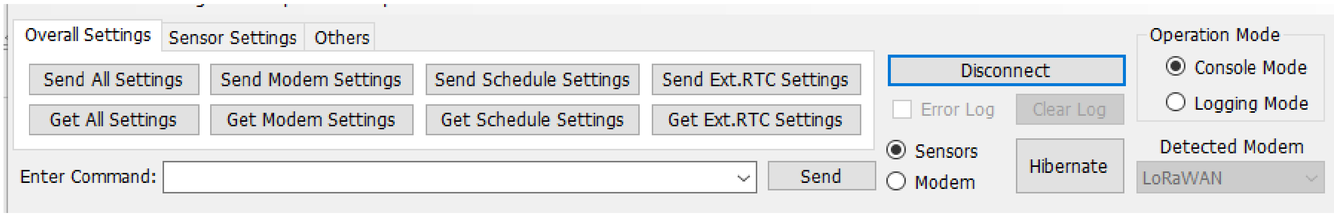
4.8.2 Manual

This opens the user manual.

4.9 Command buttons.

The main screen provides some buttons for sending parameters to the bridge and read them back. Operation mode will be disabled until COM Port is open (this is automatically done once USB connection is detected).





Operation Mode: ● Console Mode ○ Logging Mode

● Sensors ○ Modem

Detected Modem: LoRaWAN

Fig.25 Enable/Disable configuration buttons when COM Port is open/closed.

These configuration buttons are only accessible in console mode.

4.9.1 Overall Settings

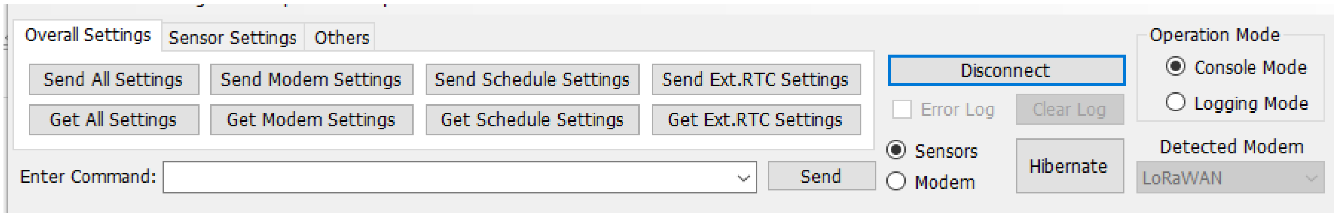


Fig.26 Overall Settings buttons.

- "Send all Settings" button: send all configuration parameters to the platform (modem, system and sensors settings).
- "Get All Settings" button: get all parameters from the platform.
- "Send LoRa Settings" button: send only LoRaWAN parameters to the platform.
- "Get LoRa Settings" button: get only LoRaWAN parameters from the platform.
- "Send Schedule Settings" button: send logging mode schedule parameters to the platform.
- "Get Schedule Settings" button: get logging mode schedule parameters from the platform.
- "Send Ext.RTC Settings" button: update TBSL1 external RTC time.
- "Get Ext.RTC Settings" button: get TBSL1 external RTC time from external RTC.

4.9.2 Sensor Settings

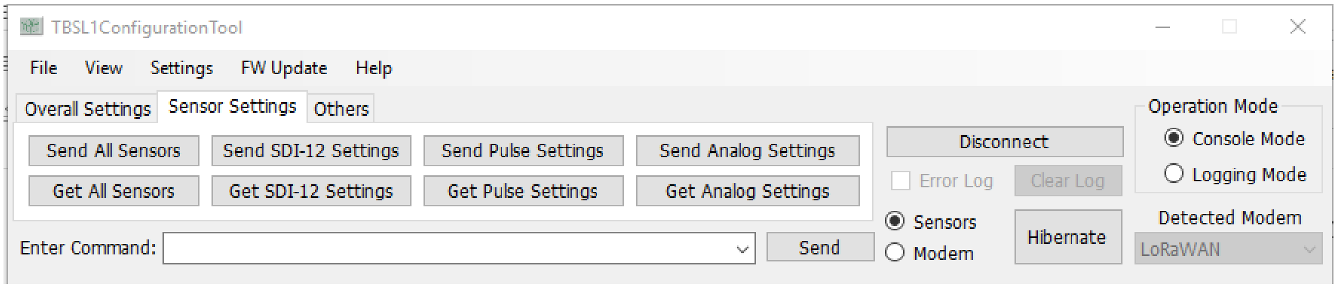


Fig.27 Sensor Settings buttons.

- "Send all Sensor" button: send all sensor settings (SDI-12, Pulse and Analog) to the platform.

- “Get all Sensor” button: get all sensor settings (SDI-12, Pulse and Analog) from the platform.

- “Send SDI-12 Settings” button: When the platform operates in console mode, this button is used to send SDI-12 sensor settings to the platform.

- “Get SDI-12 Settings” button: get SDI-12 sensor settings from the platform.

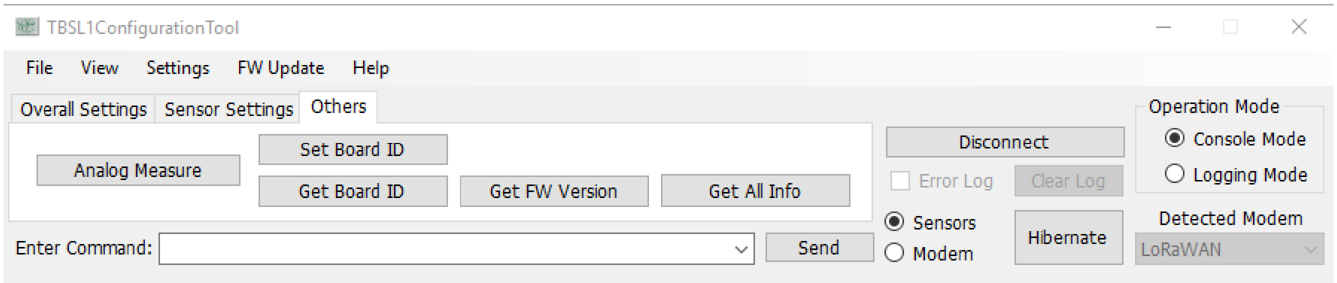
- “Send Pulse Settings” button: send pulse sensor settings to the platform.

- “Get Pulse Settings” button: get pulse sensor settings from the platform.

- “Send Analog Settings” button: send analog sensor settings to the platform.

- “Get Analog Settings” button: get analog sensor settings from the platform.

4.9.3 Others



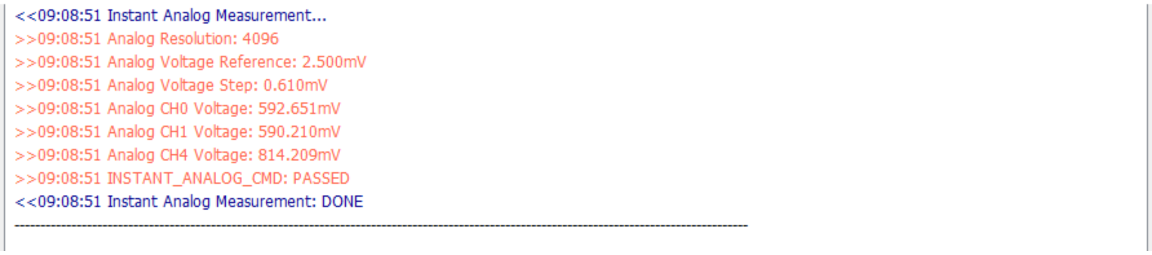
Operation Mode: ● Console Mode ○ Logging Mode

● Sensors ○ Modem

Detected Modem: LoRaWAN

Fig.28 Others buttons.

- “Analog Measure” button: get an instant measurement on all configured analog channels.



```
<<09:08:51 Instant Analog Measurement...
>>09:08:51 Analog Resolution: 4096
>>09:08:51 Analog Voltage Reference: 2.500mV
>>09:08:51 Analog Voltage Step: 0.610mV
>>09:08:51 Analog CH0 Voltage: 592.651mV
>>09:08:51 Analog CH1 Voltage: 590.210mV
>>09:08:51 Analog CH4 Voltage: 814.209mV
>>09:08:51 INSTANT_ANALOG_CMD: PASSED
<<09:08:51 Instant Analog Measurement: DONE
--------------------------------------------------------------------------------------------------------
```

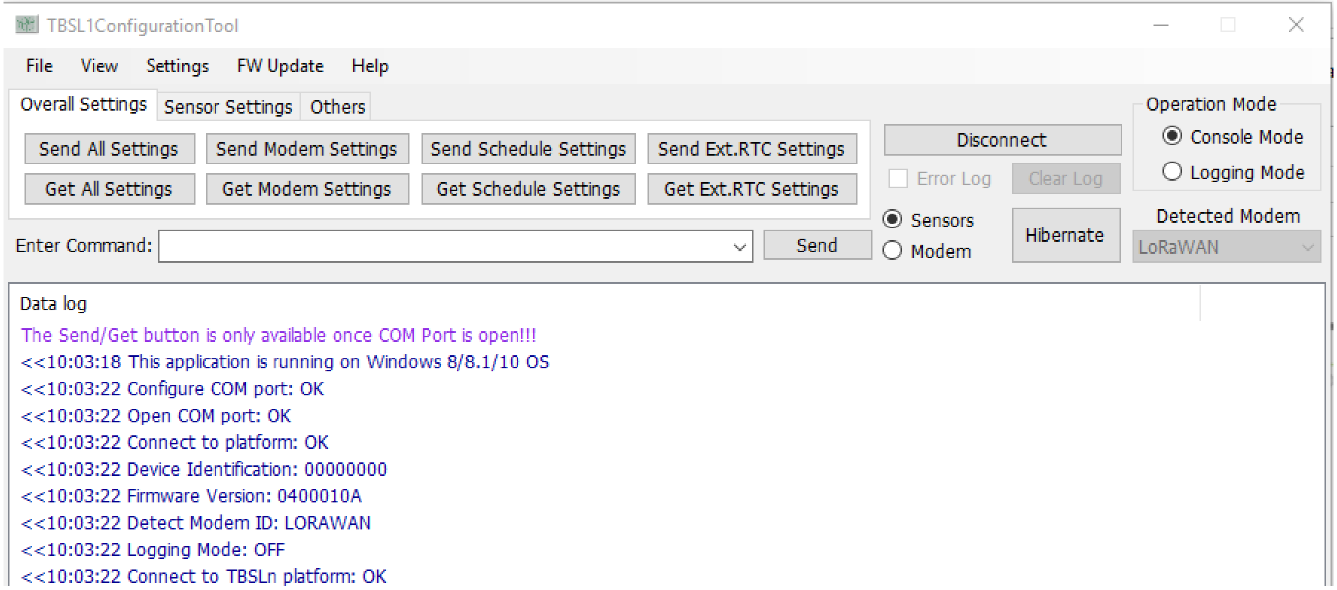
Fig.29 Instant Analog Measurement.

- “Get FW Version” button: get current TBSL1 firmware version.
- “Set Board ID” button: set TBSL1 board ID.
- “Get Board ID” button: get TBSL1 board ID.

- “Get All Info” button: get both TBSL1 firmware version and board ID.

4.10 Open Connection and Send Command

To interact with the platform, the virtual COM port must be open by the application (which is automatically done upon USB connection to the platform). This might take few seconds before connection is detected and established. After that, the GUI will send a command to check connection and get the time of TBSL1 external RTC.



```
The Send/Get button is only available once COM Port is open!!!
<<10:03:18 This application is running on Windows 8/8.1/10 OS
<<10:03:22 Configure COM port: OK
<<10:03:22 Open COM port: OK
<<10:03:22 Connect to platform: OK
<<10:03:22 Device Identification: 00000000
<<10:03:22 Firmware Version: 0400010A
<<10:03:22 Detect Modem ID: LORAWAN
<<10:03:22 Logging Mode: OFF
<<10:03:22 Connect to TBSLn platform: OK
```

Fig.30 Notifications after opening COM Port.

Once the application has opened the COM port connection, it can communicate with the platform. In console mode, the platform doesn’t perform any measurement/transmission operations. It only receives the commands from the application, executes the commands on sensors and LoRaWAN modem and send responses to the application.

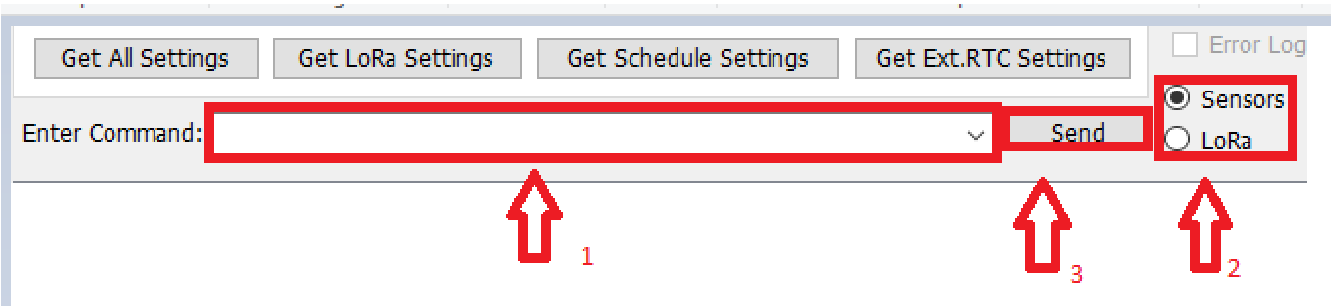


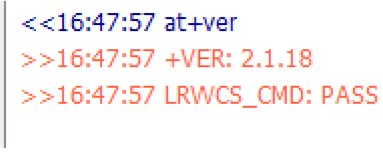
Fig.31 Send Command.

Steps required to send commands to SDI-12 sensors or LoRaWAN modem in console mode:

1. Type the command that you want to send (make sure that command’s syntax is correct)
2. Choose command’s target (SDI-12 sensor or LoRa module).

3. Click “Send command to” button.

On the data log area, transmit command strings are prefixed with “>>” whereas receive command string is prefixed with “<<” (from platform point of view).
For example to check LoRaWAN modem FW version:



```
<<16:47:57 at+ver
>>16:47:57 +VER: 2.1.18
>>16:47:57 LRWCS_CMD: PASS
```

Time stamp information is also displayed when “Tag time” option has been ticked in the “View” menu.

4.11 Operation Mode

Platform runs with 2 different operation modes: console and logging mode.

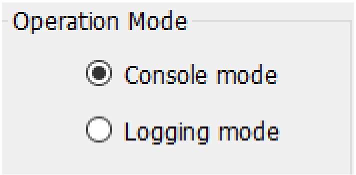
Console mode: this is configuration mode

- TBSL1 can be fully configured through the PC application
- System, modem, SDI-12, analog and pulse sensors parameters can be set
- Commands can be sent straight to modem and SDI-12 sensors
- TBSL1 FW is only monitoring commands from the PC application and sending back responses
- TBSL1 power management is disabled

Logging mode: this is operating mode

- TBSL1 parameters are not accessible anymore
- TBSL1 application FW is running (i.e. measuring, logging and transmitting)
- Power management is enabled

Switching between both mode can only be done through the PC configuration tool in Operation Mode menu:

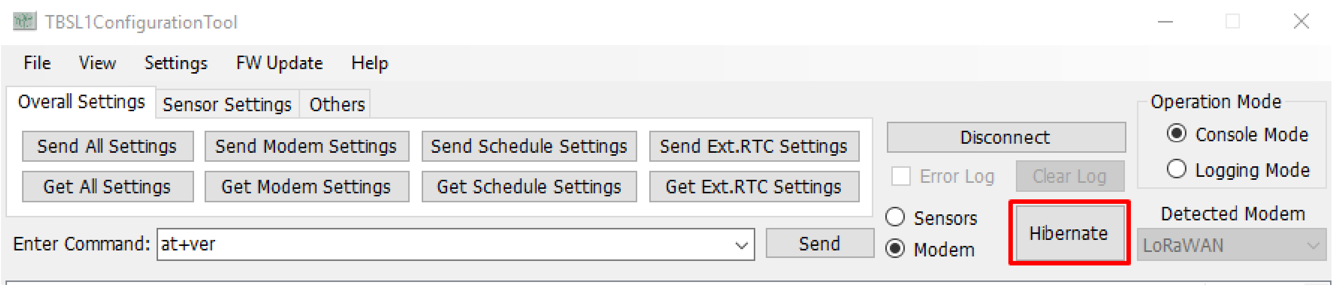


Operation Mode
● Console mode
○ Logging mode

Fig.32 Switch Mode.

The switch from Logging mode to Console mode happens in TBSL1 only when the PC application is in console mode and the platform wakes up from sleep mode.

After configuring the platform, users usually leave it on the shelves for a while before deployment.
To save power, it is then needed once configuration is completed to press the “Hibernate” button so the platform enters in deep sleep mode.



Enter Command: at+ver

○ Sensors
● Modem

☐ Error Log

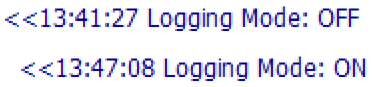
Operation Mode
● Console Mode
○ Logging Mode

Detected Modem: LoRaWAN

From that point, solar panel can also be disconnected if needed: this will power off the platform (given USB cable is unplugged).

When powered again, the platform will automatically start in logging mode.

Important: The TBSL1 platform will not switch back to Console under any circumstance except using properly switching mode function. So in some case, especially when updating to a newer version platform's FW or GUI, *if user forgets to switch both platform and GUI into Console mode before update,* it might cause a mode mismatch between GUI and platform. Normally, there is a notification in the GUI show what mode both GUI and platform are in, but if the mismatch issue happens that notification will only show current mode of the GUI.



```
<<13:41:27 Logging Mode: OFF
<<13:47:08 Logging Mode: ON
```

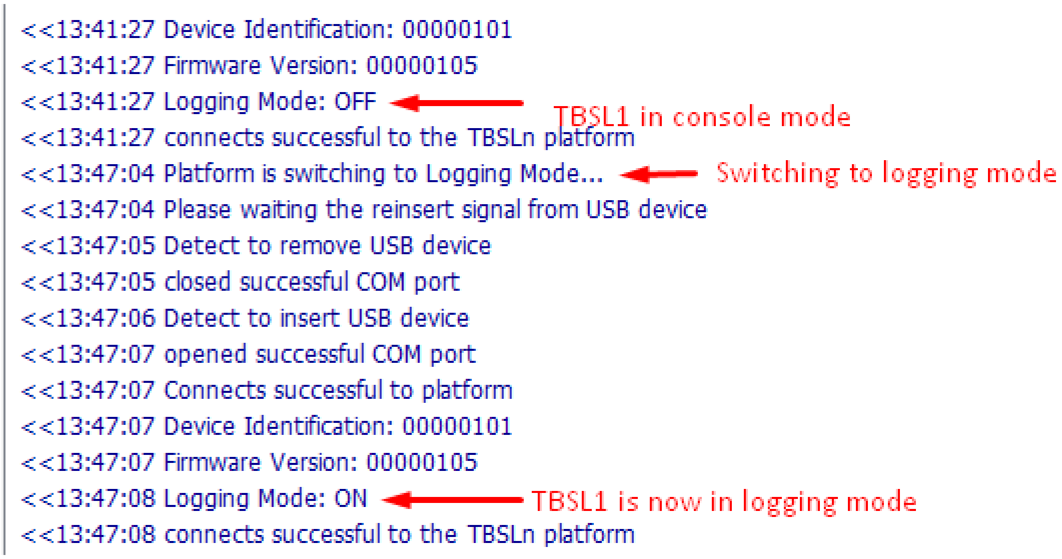
Fig.33 Mode Notification (console & logging)

Fixing mode mismatch issue: To fix this, user must follow these steps:

- Step 1: Look for the mode of the platform, by checking in the server.
    - *For instance if server is receiving data packet, the platform is in logging mode.*
- Step 2: Close the GUI if it still opened, then open it again, and DON'T CLICK ON OPEN COM yet.
- Step 3: Click on the mode matching with the current platform's mode.
    - *Continue with the example in Step 1, click on Logging Mode button.*
- Step 4: Open the COM Port.
- Step 5: Switch to opposite mode and wait for successfull switching signal. Then both platform and GUI now are in same mode.
    - *Continue with the example in Step 3, click on Console Mode button.*

Normally, the application always stays in **"Console mode"**. In this mode, the platform will be waiting to send/get parameters to/from the GUI. All the setting sent from the GUI at this time will be configured and saved in the platform memory; they are ALREADY set to the platform.
When user clicks on **"Logging mode"** button, the application sends a command to the platform to make it switch to **"Logging mode"**. The GUI notifies then the user to wait until the switch is completed *(user must not click on any button or close the GUI, else it will cause the mismatch issue*

between the GUI and the platform). It works likewise when switching from **“Logging Mode”** to **“Console Mode”**.



```
<<13:41:27 Device Identification: 00000101
<<13:41:27 Firmware Version: 00000105
<<13:41:27 Logging Mode: OFF
<<13:41:27 connects successful to the TBSLn platform
<<13:47:04 Platform is switching to Logging Mode...
<<13:47:04 Please waiting the reinsert signal from USB device
<<13:47:05 Detect to remove USB device
<<13:47:05 closed successful COM port
<<13:47:06 Detect to insert USB device
<<13:47:07 opened successful COM port
<<13:47:07 Connects successful to platform
<<13:47:07 Device Identification: 00000101
<<13:47:07 Firmware Version: 00000105
<<13:47:08 Logging Mode: ON
<<13:47:08 connects successful to the TBSLn platform
```

TBSL1 in console mode

Switching to logging mode

TBSL1 is now in logging mode

Fig.34 Switch to Logging Mode.

After that, the platform will go to sleep. From here onwards, user has two options:

1. Switch to **“Console mode”** by clicking on **“Console mode”** button. This sends a command to the platform so whenever it wakes up, it receives that command and switch to **“Console mode”**.
2. Stay in **“Logging mode”**. In this state after wake up, the platform will start the regular sequence of *measure data* -> *go to sleep* -> *wake up* -> *measure again and/or transmit data*
   - ➔ *go to sleep again and go on*. The time between each sleep and wake up depend on the Interval settings in the application.

Note: even if user clicks “Console mode” while the platform is still self-configuring its internal settings or measuring/transmitting data to server, the platform will only switch back to “Console mode” button after it wakes up.

4.12 Data format

Several reporting messages are transmitted by TBSL1:

- Common data message
  - Ex: ***C00004AC10400010b0fR -53***
  - Fields:
    - Report ID: Message for common data reporting. Value of this field is character ‘C’.
    - 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”

- Battery report message
  - Ex: ***PB16:10:27:14:45:00* 4.100**
  - 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: parameter of measured pulses 4.100.

- Pulses report message
  - Ex: ***PP16:10:27:02:45:000* 12004**
  - Fields:
    - **P:** parameters report message
    - **P:** pulse sensor
    - **16:10:27:02:45:00** time stamp, 2016 October 27th 02:45:00AM (HH:MM:SS)
    - **0:** pulse sensor ID
    - **<space>**
    - **12004:** pulse count
- Analog report message
  - Ex: ***PA16:10:29:14:30:001* 0.235532 3.454323 5.454323**
  - Fields:
    - **P:** parameters report message
    - **A:** analog sensor
    - **16:10:29:14:30:00** time stamp, 2016 October 29th 14:30:00 (ie 2:30:00 PM, HH:MM:SS)
    - **<space>**
    - **0.235532:** minimum measured analog value
    - **<space>**
    - **3.454323:** average of analog measured values
    - **<space>**
    - **5.454323:** maximum measured value

- SDI-12 report message
  - Ex: ***PS16:10:29:14:30:000304 -12.20003 +2.322432 -4.433332 -9.110423***
  - 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”.

Refer to TBSL1-N datasheet for further description of each message format.

5 History

Version	Date	Author	Changes
V1.0	18/05/2017	Tuan Anh	Update from TBSL1 DemoFW Document
V1.1	24/05/2017	Tuan Anh	Update some important notifications
V1.2	27/06/2017	Tuan Anh	Add FW Update feature.
V1.3	14/09/2017	Philippe	Update GUI screenshots and some comments.
V1.4	19/06/2018	Philippe	Update GUI screenshots and some comments/Hibernate feature/syntax corrections
V1.5	14/11/2018	Philippe	Update DFU FW update procedure / Add cellular modem configuration.