DESCRIPTION

TESNLTB is a LoRA® / LoRaWan™ sensor board designed for industrial remote monitoring, schools, education centres and laboratories to quickly develop IoT sensing applications.

The board integrates a complete RF hardware and software LoRaWan system, SMA connector for antenna connection, long range external antenna, high performance low power MCU, many controllable mems and ambient environmental sensors.

It can be fully controlled and re-configured remotely, allowing flexible management of the board operating in remote locations.

FEATURES

- LoRa / LoRaWan, Class A end device
- MAC V1.0.2, PHY V1.0.2 REV B

- Integrated SoC with Semtech SX1276 LORA chip, RF frontend and STM32L073Z MCU

- Powered by 3 x AA batteries

- Accelerometer & Gyroscope sensors

- Magnetometer sensor

- Temperature & Humidity sensor

- Ambient audio noise level microphone

- Ambient light & Proximity sensor

- Pressure & Altitude sensor

- 4 x GPIOs & 2 x ADC inputs

- LED indicator controllable from gateway

- Dimension : 80mm x 53mm x 32mm (excluding antenna)

- Server-side demo source code and application notes available

ORDERING INFORMATION

Model	Description	Region
TESNLTB-AS	LoRa / LoRaWan IoT Sensor Node, 923-925Mhz	Asia
TESNLTB-EU	LoRa / LoRaWan IoT Sensor Node, 868Mhz	Europe
TESNLTB-US	LoRa / LoRaWan IoT Sensor Node, 915Mhz	USA

PACKAGE CONTENTS

- TESNLTB LoRA® / LoRaWan™ sensor board endpoint
- External detachable antenna

ELECTRICAL SPECIFICATION

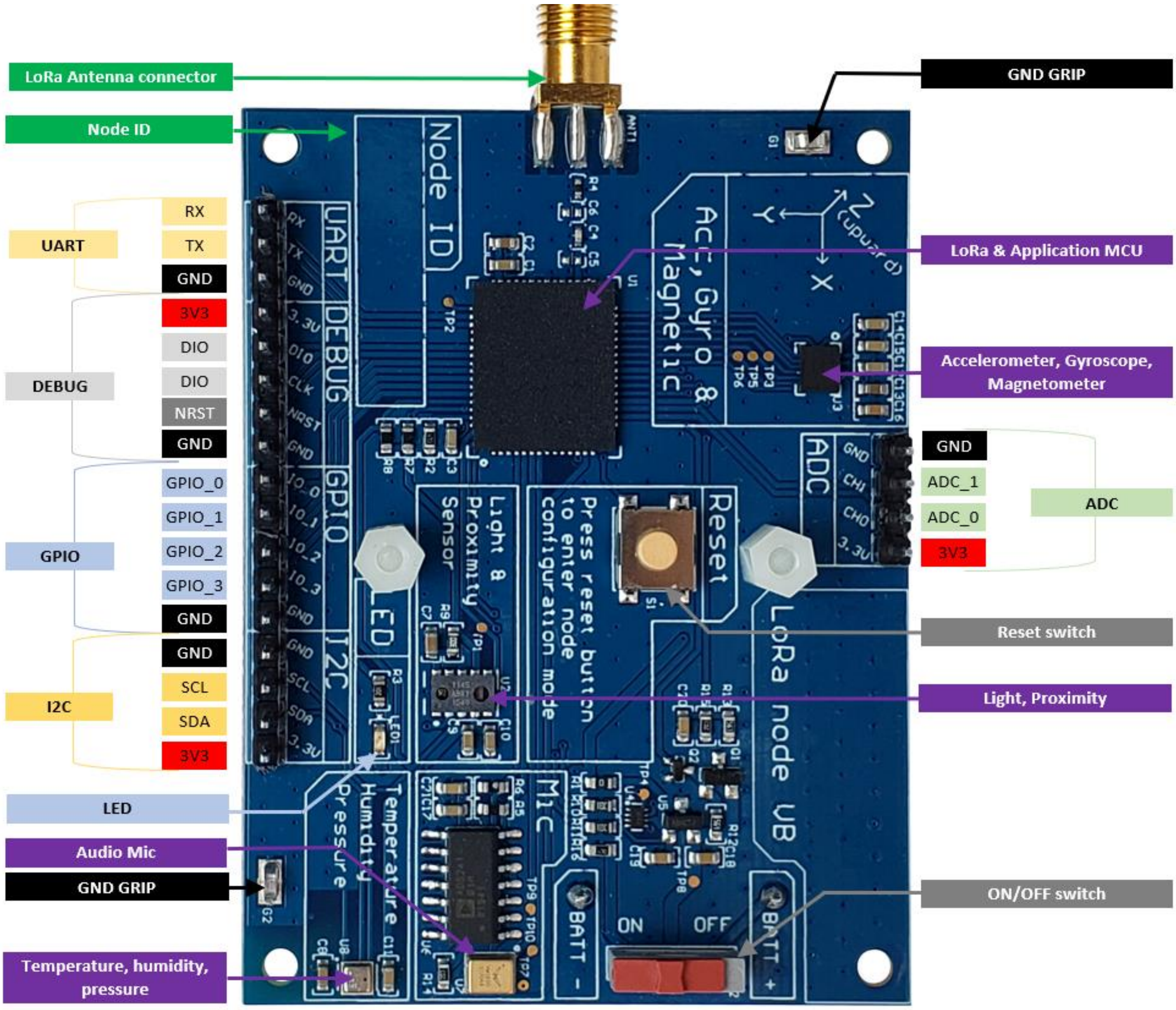
Parameter	Min	Typical	Max	Unit	Remark
Power supply					
Power Supply voltage	2.6	4.5	5	V	
Standby mode current		1.7	5	uA	
System operating current		9	9.8	mA	
Vdd (generated internally by LDO)		3.3		V	
Radio					
LoRa Tx +20dB +17dB +13dB +7dB		 127 82 65 49		mA	
LoRa Rx		17.5		mA	
Frequency range	863	915	928	MHz	
SMA Antenna jack impedance		50		Ohm	
SMA antenna jack diameter		10		mm	
SMA antenna jack length		108		mm	
Sensor power consumption					
Temperature sensor measuring mode current		350		uA	
Pressure sensor measuring mode current		714		uA	
Humidity sensor measuring mode current		340		uA	
Temperature, pressure and humidity sensor measuring duration	100		200	us	
Light sensor measuring current		4.8		mA	
Light sensor measuring duration	200		600	us	
accelerometer and magnetometer operating current when enabled		600		uA	Always on if enabled
Gyroscope operating current when enabled		4		mA	
Noise sensor operating current		100		uA	
I/O port (GPIO, UART)					
Input low level voltage	0		0.99	V	
Input high level voltage	2.31		3.3	V	
Output low level voltage		0		V	
Output high level voltage		3.3		V	
Output sunk current			16	mA	
Output source current	-16			mA	
Onboard LED current		5.5		mA	
UART					
Bit rate		115200		bps	
Data bit		8		bit	
stop bit		1		bit	

SENSOR CHARACTERISTICS

Parameter	Min	Typical	Max	Unit	Remark
Temperature					
Sensor range	-40	25	85	°C	
Sensor accuracy		±0.5	±1.5	°C	
Resolution		0.01		°C	
Humidity					
Sensor range	0		100	%	
Sensor accuracy		±3		%	
Resolution		0.008		%	
Pressure & Altitude					
Sensor range	300		1100	hPa	
Sensor accuracy		±1	±1.7	hPa	
Resolution		0.18		Pa	
Accelerometer					
Axis		3			(x,y,z)
Full scale		±4		g	±2/±8/±16 also available (1)
Sensitivity		0.122		mg	
Gyroscope					
Axis		3			(x,y,z)
Full scale		±2000		dps	±245/±500 also available (1)
Sensitivity		70		mdps	
Magnetometer					
Axis		3			(x,y,z)
Full scale		±16		gauss	±4/±8/±12 also available (1)
Sensitivity		0.58		mguass	
Microphone					
Range	65		89	dBA	Average noise
Ambient light					
Full scale			128000	lux	
Resolution			0.1	lux	
Sensitivity	1		1.95	lux	
Proximity Sensor					
Detection range	1		50	cm	
ADC					
Measurement range	0		3.3	V	
Resolution			0.8057	mV	4096 steps

(1) For Accelerometer, Gyrocope and magnetometer, different full-scale range can be provided with firmware upgrade, please contact our sale representative for detail.

PIN ASSIGNMENT



LoRaWan PROVISIONING

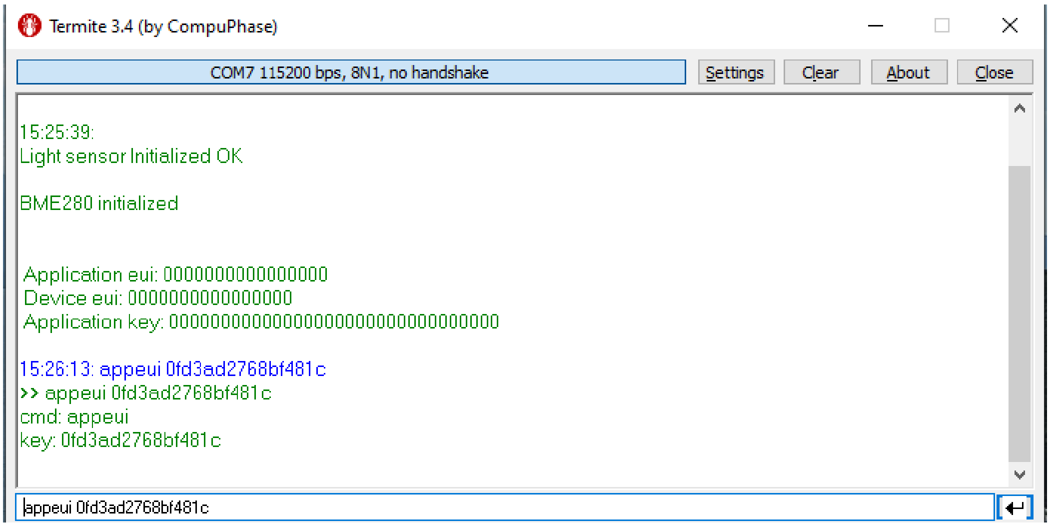
LoRaWan provisioning is performed by UART.

The provisioning consists in providing the AppEUI, devEUI and appKey to the board. The firmware will then save these parameters in the MCU's EEPROM and can reload them automatically at boot.

1. Turn ON the board and connect the UART (GND, RX and TX) to a computer using a USB to **3.3V UART adapter**.

2. Open UART with speed 115200bps, 8 data bits and 1 stop bit.

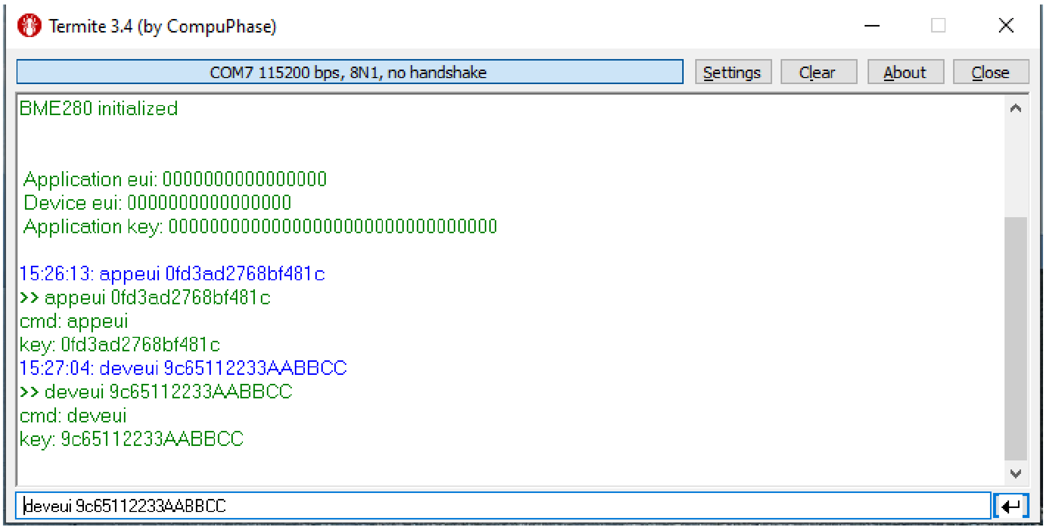
3. Send appEUI command followed by 8 bytes hex application eui:



```
15:25:39:
Light sensor Initialized OK

BME280 initialized


 Application eui: 0000000000000000
 Device eui: 0000000000000000
 Application key: 00000000000000000000000000000000

15:26:13: appeui 0fd3ad2768bf481c
>> appeui 0fd3ad2768bf481c
cmd: appeui
key: 0fd3ad2768bf481c
```

```
appeui 0fd3ad2768bf481c
```

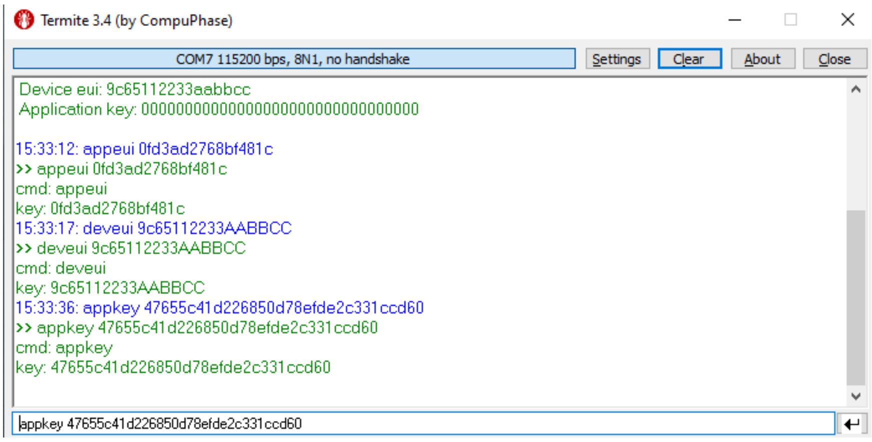
4. Send devEUI command followed by 8 bytes hex device EUI.



```
BME280 initialized


 Application eui: 0000000000000000
 Device eui: 0000000000000000
 Application key: 00000000000000000000000000000000

15:26:13: appeui 0fd3ad2768bf481c
>> appeui 0fd3ad2768bf481c
cmd: appeui
key: 0fd3ad2768bf481c
15:27:04: deveui 9c65112233AABBCC
>> deveui 9c65112233AABBCC
cmd: deveui
key: 9c65112233AABBCC
```

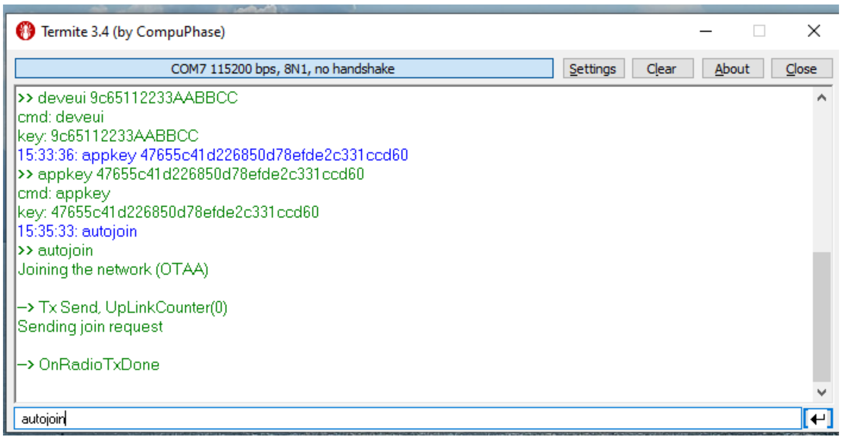
```
deveui 9c65112233AABBCC
```

5. Send appKey command followed by 16 bytes hex application key



```
COM7 115200 bps, 8N1, no handshake
Device eui: 9c65112233aabbcc
Application key: 00000000000000000000000000000000

15:33:12: appeui 0fd3ad2768bf481c
>> appeui 0fd3ad2768bf481c
cmd: appeui
key: 0fd3ad2768bf481c
15:33:17: deveui 9c65112233AABBCC
>> deveui 9c65112233AABBCC
cmd: deveui
key: 9c65112233AABBCC
15:33:36: appkey 47655c41d226850d78efde2c331ccd60
>> appkey 47655c41d226850d78efde2c331ccd60
cmd: appkey
key: 47655c41d226850d78efde2c331ccd60

appkey 47655c41d226850d78efde2c331ccd60
```

6. Enable autojoin to let the board automatically connect to the LoRa network at boot.



```
COM7 115200 bps, 8N1, no handshake
>> deveui 9c65112233AABBCC
cmd: deveui
key: 9c65112233AABBCC
15:33:36: appkey 47655c41d226850d78efde2c331ccd60
>> appkey 47655c41d226850d78efde2c331ccd60
cmd: appkey
key: 47655c41d226850d78efde2c331ccd60
15:35:33: autojoin
>> autojoin
Joining the network (OTAA)

--> Tx Send, UpLinkCounter(0)
Sending join request

--> OnRadioTxDone

autojoin
```

Supported UART commands:

UART Command	Description
`appeui [hex app eui]`	Use this command to input and store the application EUI to the device for joining the LoRa network.
`deveui [hex device EUI]`	Use this command to input and store the device EUI to the device for joining the LoRa network.
`appkey [hex application key]`	Use this command to input and store the application key to the device for joining the LoRa network.
`autojoin`	Start joining the LoRa network with the stored provisioning information. Then set the device to automatic join the network every time after reset.
`join`	Start joining the LoRa network with the stored provisioning information and configure the device not to automatically join the network after reset.

FIRMWARE GENERAL LOGIC FLOW

SENSOR NODE CONFIGURATION

The 12 sensors in sensor board endpoint can be configured independently, they are grouped into 3 groups (3 “ports” in LoRa’s terminology). Sensor within the same group will make measurement at the same time and send the measurement result within the same uplink message. User can assign different measurement interval for each port.

A Lora uplink will be sent from the sensor node board to the server after each scheduled measurement. User can update the configuration only following an uplink. Which means that if the configured measurement interval is 1 day, it can take at most 1 day for the change to be effective. If user wants to change the configuration immediately, he can reset the sensor node by pressing the reset button on the sensor node.

Beside the periodic scheduled measurement, the sensor node can also work as threshold-based alarm (for all MEMS sensors on port 3 and some GPIO on port 4). User can configure a trigger threshold for those sensors capable of trigger alarm and whenever the sensor detects an over-threshold event, an uplink of alarm will be sent to the server immediately.

The alarm system and the periodic schedule system works independently, which means that the alarm can still be triggered even when the periodic sensing is configured to “OFF” or the interval measurement of the corresponding port is set to “NEVER”.

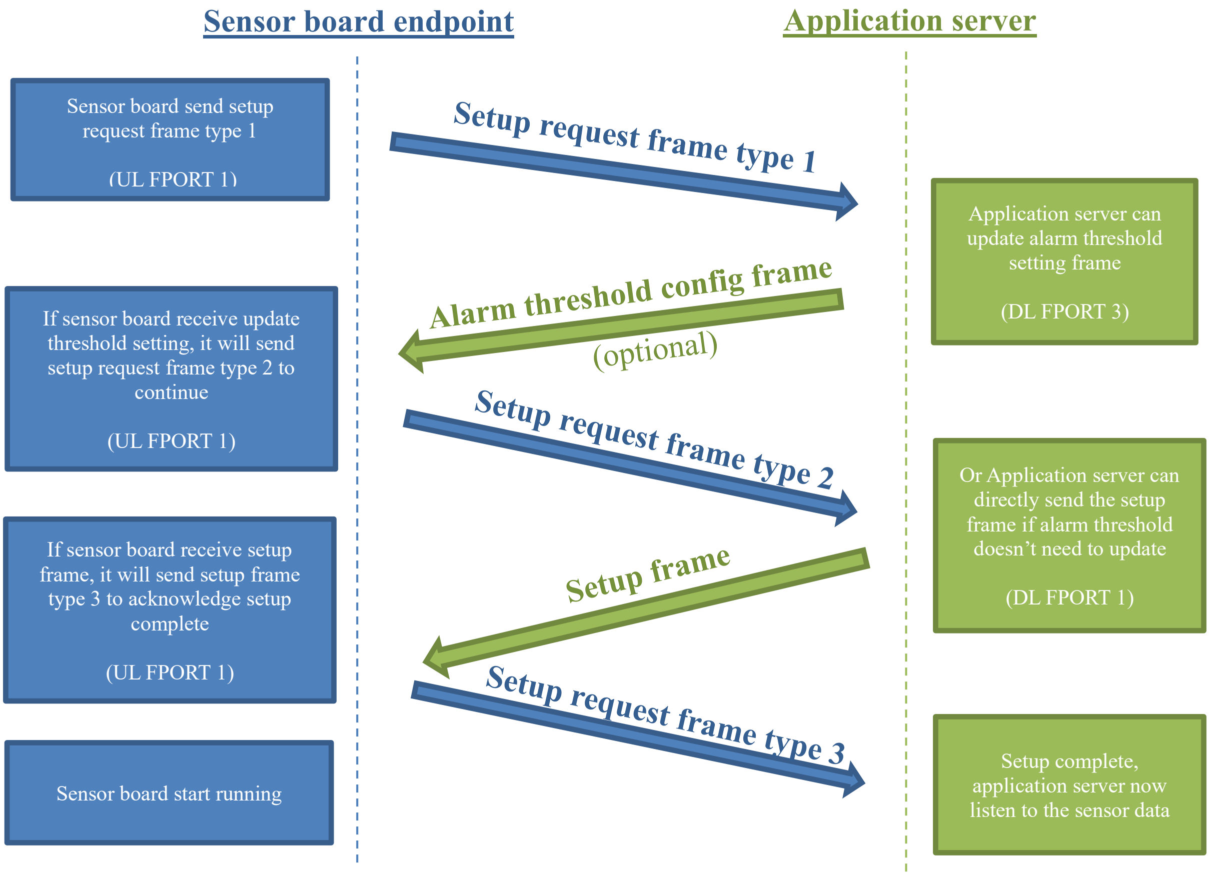
Port	Sensor	Periodic Measurement intervals	Can triggering alarm
Port 2 Environmental sensor port	Temperature	Configurable intervals: • Continuous • Every 5 secs • Every 30 secs • Every 1 min • Every 10 mins • Every 30 mins • Every 1 hour • Every 3 hours • Every 12 hours • Every 1 day • Never	
	Humidity		
	Pressure		
	Light		
Port 3 MEMS sensor port	MIC (noise level)		✓
	Accelerometer		✓
	Gyroscope		✓
	Magnetometer		✓
Port 4 User interface port	ADC x 2		
	GPIO x 4		GPIO 2 & 3
	Proximity sensor		

COMMUNICATION PROTOCOL

Sensor board endpoint communicates with LoRa network server by LoRaWan protocol. Communication starts with the sensor node sending a join request to the server/gateway. Once the join request is accepted, the sensor node will become part of the network. The server/gateway will be responsible for defining the link parameter, such as the data rate / TxRx timing, etc.

Sensor board endpoint is a class A device which means it will be always initiating the communication(uplink), and the server can only response with a downlink following an uplink. To interact with the sensor board endpoint, user need to define an application server or relay the message to an application such as by MQTT. (For detail on how to create application server you can refer to the gateway of your choice or refer to our MQTT application example).

The first uplink message the sensor board endpoint sends after it joins a network is a configuration request at LoRa uplink frame port 1 (UL FPORT 1). And the application server will response the configuration to the sensor board. Both the sensor board and the application will use the frame port number to differentiate the message type. Complete hand shaking and data format as below:



The application server can also send the update alarm threshold setting frame or the setup frame to the sensor board endpoint as a downlink message after any other uplink messages when the setting needs to be changed, and the setup process will be the same as above.

After the sensor board endpoint is running, whenever it finished a scheduled measurement, it will send the sensor measurement data at uplink port2, port3 or port4 according to the sensor group or at uplink port5 if it is alarm event.

Application server can also send control message to the sensor board endpoint at downlink port2 to control the four GPIO on the sensor port as output port or turn on/off the LED on the sensor board.

LIST OF POSSIBLE MESSAGES

UPLINK (from sensor board to server)		
Frame port #	Message type	Data length
FPORT 1	Setup request frame (type 1, 2 or 3)	4
FPORT 2	Environmental sensor measurement data	14
FPORT 3	MEMS sensor measurement data	20
FPORT 4	User interface input data	7
FPORT 5	Alarm trigger event data	8

DOWNLINK (from server to sensor board)		
Frame port #	Message type	Data length
FPORT 1	Setup frame	6
FPORT 2	Sensor board output control frame	2
FPORT 3	Alarm threshold configuration frame	6

UPLINK MESSAGE FORMAT

UPLINK FPORT 1 (SETUP REQUEST FRAME) (4 BYTES)	
Byte 0	Sensor board version number first digit (x in x.y.z)
Byte 1	Sensor board version number second digit (y in x.y.z)
Byte 2	Sensor board version number third digit (z in x.y.z)
Byte 3	Setup request frame type 1: Requesting alarm threshold configuration 2: Acknowledge alarm threshold received, requesting sensor board setup 3: Acknowledge setup frame received, setup complete

UPLINK FPORT 2 (ENVIRONMENTAL SENSOR PORT) (14 BYTES)	
Byte [0 – 3]	Temperature sensor reading (Little endian) T = value / 100 (°C)
Byte [4 – 7]	Humidity sensor reading (Little endian) H = value / 1024 (%)
Byte [8 – 11]	Pressure sensor reading (Little endian) P = value / 10000 (kPa)
Byte [12 – 13]	Light sensor reading (Little endian) L = value - 255 (lux)

UPLINK FPORT 3 (MEMS SENSOR PORT) (20 BYTES)	
Byte [0 – 1]	Accelerometer sensor reading (x-axis) (Little endian) Accx = value * 0.122 (mg)
Byte [2 – 3]	Accelerometer sensor reading (y-axis) (Little endian) Accy = value * 0.122 (mg)
Byte [4 – 5]	Accelerometer sensor reading (z-axis) (Little endian) Accz = value * 0.122 (mg)
Byte [6 – 7]	Gyroscope sensor reading (x-axis) (Little endian) Gyrox = value * 0.07 (DPS)
Byte [8 – 9]	Gyroscope sensor reading (y-axis) (Little endian) Gyroy = value * 0.07 (DPS)
Byte [10 – 11]	Gyroscope sensor reading (z-axis) (Little endian) Gyroz= value * 0.07 (DPS)
Byte [12 – 13]	Magneto sensor reading (x-axis) (Little endian) Magx = value * 0.58 (mgauss)
Byte [14 – 15]	Magneto sensor reading (y-axis) (Little endian) Magy = value * 0.58 (mgauss)
Byte [16 – 17]	Magneto sensor reading (z-axis) (Little endian) Magz= value * 0.58 (mgauss)
Byte [18 – 19]	Noise sensor reading (Little endian) N = value (dBA)

<table>
<tr><th colspan="7">UPLINK FPORT 4 (USER INTERFACE INPUT SENSOR PORT) (8 BYTES)</th></tr>
<tr><td>Byte [0 – 1]</td><td colspan="6">ADC0 reading (Little endian)
Voltage = value / 4096 x 3.3 (V)</td></tr>
<tr><td>Byte [2 – 3]</td><td colspan="6">ADC1 reading (Little endian)
Voltage = value / 4096 x 3.3 (V)</td></tr>
<tr><td>Byte [4 – 5]</td><td colspan="6">Proximity sensor reading (Little endian)</td></tr>
<tr><td rowspan="3">Byte [6]</td><td colspan="6">Logic level of GPIO and LED (1: high, 0: low)</td></tr>
<tr><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2 – b0</td></tr>
<tr><td>GPIO0</td><td>GPIO1</td><td>GPIO2</td><td>GPIO3</td><td>LED</td><td>X</td></tr>
</table>

<table>
<tr><th colspan="2">UPLINK FPORT 5 (ALARM TRIGGER EVENT PORT) (4 BYTES)</th></tr>
<tr><td>Byte [0 – 1]</td><td>ID indicating the triggering source
<table>
<tr><th>ID</th><th>Triggering source</th></tr>
<tr><td>1</td><td>Sound</td></tr>
<tr><td>2</td><td>Accelerometer</td></tr>
<tr><td>3</td><td>Gyroscope</td></tr>
<tr><td>4</td><td>Magnetometer</td></tr>
<tr><td>5</td><td>GPIO 3</td></tr>
<tr><td>6</td><td>GPIO 2</td></tr>
<tr><td>7</td><td>battery level</td></tr>
</table></td></tr>
<tr><td>Byte [2 – 3]</td><td>Additional information
<table>
<tr><th>if ID is</th><th>This data represents</th></tr>
<tr><td>1 - 4</td><td>No additional information</td></tr>
<tr><td>5 or 6</td><td>GPIO logic level at message send time (1: high, 0: low)
<table>
<tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3 – b0</th></tr>
<tr><td>GPIO0</td><td>GPIO1</td><td>GPIO2</td><td>GPIO3</td><td>x</td></tr>
</table></td></tr>
<tr><td>7</td><td>battery level: 1 – 254 (following LoRa specification)
1 being minimum, 254 being maximum</td></tr>
</table></td></tr>
</table>

<table>
<tr><th colspan="2">DOWNLINK FPORT 1 (SETUP FRAME) (6 BYTES)</th></tr>
<tr><td>Byte 0</td><td>Environmental sensor configuration
<table>
<tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3 – b0</th></tr>
<tr><td>Temperature Enable</td><td>Humidity Enable</td><td>Pressure Enable</td><td>Light Enable</td><td>Port 2 reporting frequency (1)</td></tr>
</table></td></tr>
<tr><td>Byte 1</td><td>MEMS sensor configuration
<table>
<tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3 – b0</th></tr>
<tr><td>Sound Enable</td><td>Accelerometer Enable</td><td>Gyroscope Enable</td><td>Magnetometer Enable</td><td>Port 3 reporting frequency (1)</td></tr>
</table></td></tr>
<tr><td>Byte 2</td><td>User interface configuration
<table>
<tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3 – b0</th></tr>
<tr><td>ADC0 Enable</td><td>ADC1 Enable</td><td>Proximity Enable</td><td>GPIO input Enable</td><td>Port 4 reporting frequency (1)</td></tr>
</table></td></tr>
<tr><td>Byte 3</td><td>Alarm setting 1
<table>
<tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3 – b0</th></tr>
<tr><td>Sound trigger Enable</td><td>Accelerometer trigger Enable</td><td>Gyroscope trigger Enable</td><td>Magnetometer trigger enable</td><td>x</td></tr>
</table></td></tr>
<tr><td>Byte 4</td><td>Alarm setting 2
<table>
<tr><th>b7 – b6</th><th>b5 – b4</th><th>b3 – b2</th><th>b1 – b0</th></tr>
<tr><td>GPIO 3 trigger setting (2)</td><td>GPIO2 trigger setting (2)</td><td>LED mode (3)</td><td>x</td></tr>
</table></td></tr>
<tr><td>Byte 5</td><td>GPIO output setting
<table>
<tr><td>b7</td><td>GPIO0 input / output mode (1: output, 0: input)</td></tr>
<tr><td>b6</td><td>GPIO0 output level (1: high, 0: low)</td></tr>
<tr><td>b5</td><td>GPIO1 input / output mode (1: output, 0: input)</td></tr>
<tr><td>b4</td><td>GPIO1 output level (1: high, 0: low)</td></tr>
<tr><td>b3</td><td>GPIO2 input / output mode (1: output, 0: input)</td></tr>
<tr><td>b2</td><td>GPIO2 output level (1: high, 0: low)</td></tr>
<tr><td>b1</td><td>GPIO3 input / output mode (1: output, 0: input)</td></tr>
<tr><td>b0</td><td>GPIO3 output level (1: high, 0: low)</td></tr>
</table></td></tr>
</table>

(1) Frequency setting options	
0	Disable sensor port
1	continuous measurement
2	every 5 seconds
3	every 30 seconds
4	every 1 minutes
5	every 10 minutes
6	every 30 minutes
7	every 1 hour
8	every 3 hours
9	every 12 hours
10	every 24 hours

(2) GPIO trigger setting options	
0	Disable interrupt
1	Rising edge trigger
2	Falling edge trigger
3	Both edge trigger

(3) LED mode setting options	
0	LED off
1	LED On
2	LED as status LED

DOWNLINK FPORT 2 (SENSOR BOARD OUTPUT CONTROL) (2 BYTES)

GPIO output setting

b7	GPIO0 input / output mode (1: output, 0: input)
b6	GPIO0 output level (1: high, 0: low)
b5	GPIO1 input / output mode (1: output, 0: input)
b4	GPIO1 output level (1: high, 0: low)
b3	GPIO2 input / output mode (1: output, 0: input)
b2	GPIO2 output level (1: high, 0: low)
b1	GPIO3 input / output mode (1: output, 0: input)
b0	GPIO3 output level (1: high, 0: low)

b7	b6 – b2	b1 – b0
Battery level check	x	LED mode

LED mode setting options	
0	LED off
1	LED On
2	LED as status LED

when the sensor board receive a battery check request from a downlink, it will perform a battery level measurement. Once the measurement is done, the result will be reported in Uplink FPORT5 as a trigger event.

DOWNLINK FPORT 3 (ALARM THRESHOLD CONFIGURATION) (6 BYTES)	
Byte 1	Sound trigger threshold, Byte 1 value: 0-255, default is 0 Threshold (dBA) = 6.4 x ln(20000 + 3920 x [Byte 1 value])
Byte 2	Accelerometer trigger threshold value: 0-255 0: Use recommended threshold of 784mG 1-255: Threshold (mG) = 15.68 x value
Byte [3 – 4]	Gyroscope trigger threshold Byte 3 & 4 in little endian, value 0-32767 0: Use recommended threshold of 183 DPS 1-32767: Threshold (mDPS) = 61 x value bit 15 will be ignored.
Byte [5 – 6]	Magnetometer trigger threshold Byte 5 & 6 in little endian, value 1-32767 0: Use recommended threshold of 2450 1-32767: Threshold(mGuass) = 0.49 x value bit 15 will be ignored.

Byte 1 lookup table:

Byte 1 value	Threshold (dBA)
0	63.4
1	65.0
3	66.5
6	68.1
9	69.7
12	71.2
17	72.8
23	74.4
31	75.9
41	77.5
54	79.1
71	80.6
92	82.2
119	83.8
153	85.4
197	86.9
255	88.5