UM2085
User manual

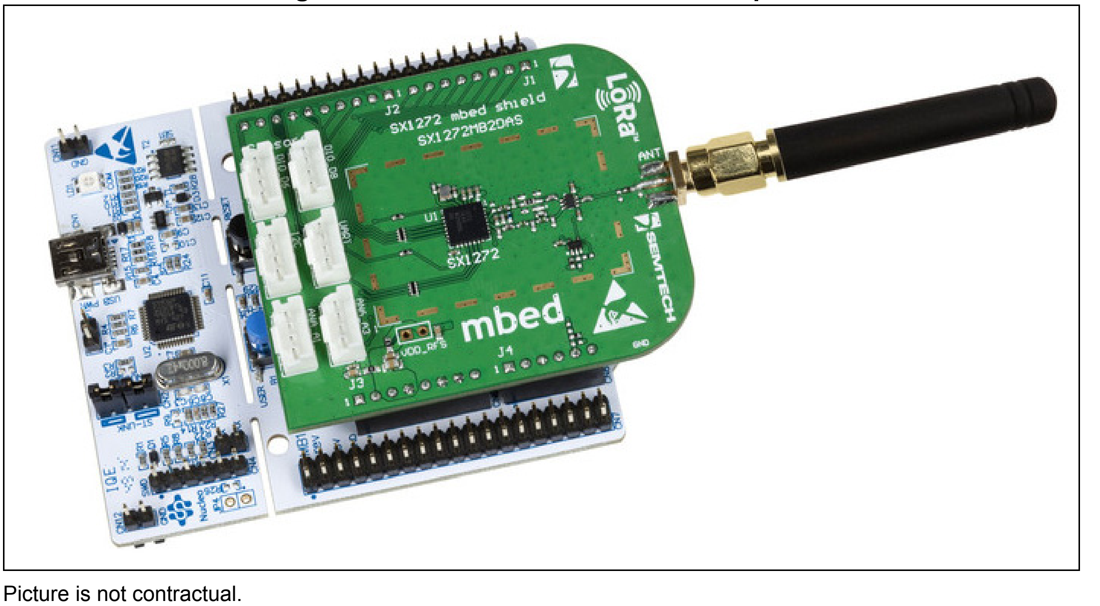
Ultra-low-power STM32 and LoRa® Nucleo pack with NUCLEO-L073RZ board and I-NUCLEO-SX1272D RF expansion board

Introduction

The ultra-low-power STM32 and LoRa® Nucleo pack (P-NUCLEO-LRWAN1) is a kit based on a NUCLEO-L073RZ board, an I-NUCLEO-SX1272D LoRa® RF expansion board from Semtech corporation, and a sub-gigahertz SMA antenna. The expansion board includes the SX1272 low-power transceiver which features the LoRa® long-range modem. This modem provides high-performance LoRa® modulation as well as OOK / FSK modulation. It is optimized for use in the 868 MHz and 915 MHz bands and its maximum output power is 14 dBm. The P-NUCLEO-LRWAN1 Nucleo pack is compatible with the I-CUBE-LRWAN Expansion Package -a certified middleware stack- that is compliant with the LoRaWAN™ V1.0.2 specification. It provides support for bidirectional end-devices in Class-A, Class-B and Class-C protocols and for end-device activation either through over-the-air activation (OTAA) or activation by personalization (ABP).

After briefly introducing end-to-end operations, this document describes the main components of the P-NUCLEO-LRWAN1 Nucleo pack and how to configure them to join and participate in a LoRa® network.

Figure 1. P-NUCLEO-LRWAN1 Nucleo pack



Picture is not contractual.

Contents

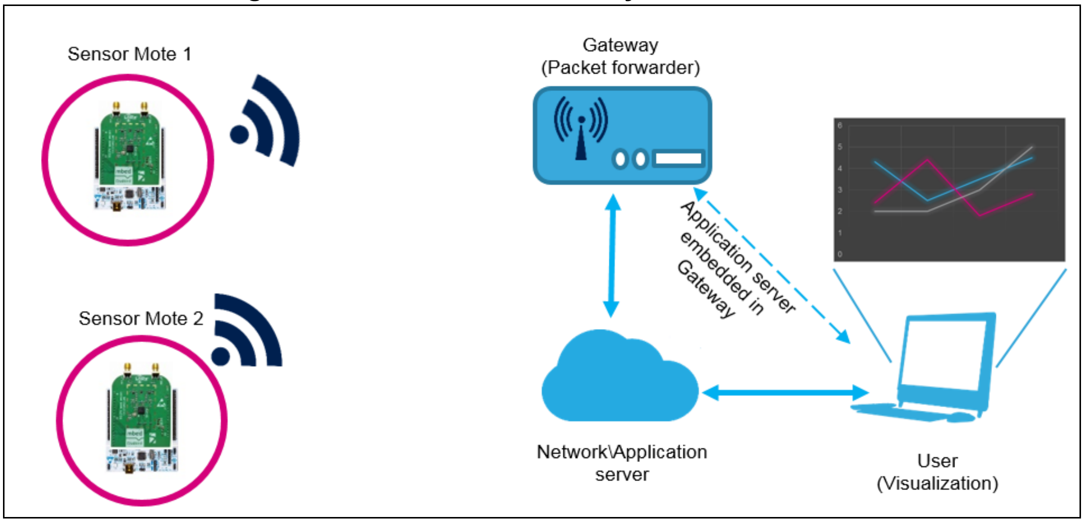
List of tables

List of figures

1 LoRa overview

The *Figure 2* shows an overview of the LoRa end-to-end link. Motes are also called nodes or end-devices. The end-devices collect data provided by environmental sensors like temperature, humidity and pressure, then send the sensor data to the gateway through an RF LoRa link. The gateway acts as a packet forwarder to the network and application server, where sensor data are retrieved and post-processed.

Figure 2. P-NUCLEO-LRWAN1 system architecture



A gateway in a final environment acts as a packet forwarder, meaning that it forwards the application payload to the network/application server using 3G/Ethernet backhaul. However, for development purpose, the gateway can be programmed to embed an Application server and managed locally (see the dashed arrow in *Figure 2*).

2 System description

This section describes the hardware components needed to build the end node and prepare the gateway for an end-to-end communication.

The hardware components are:

- NUCLEO-L073RZ board (see *Section 2.1: NUCLEO-L073RZ board*)
- I-NUCLEO-SX1272D expansion board (see *Section 2.2: I-NUCLEO-SX1272D expansion board*)
- Gateway (see *Section 2.3: Gateway*)

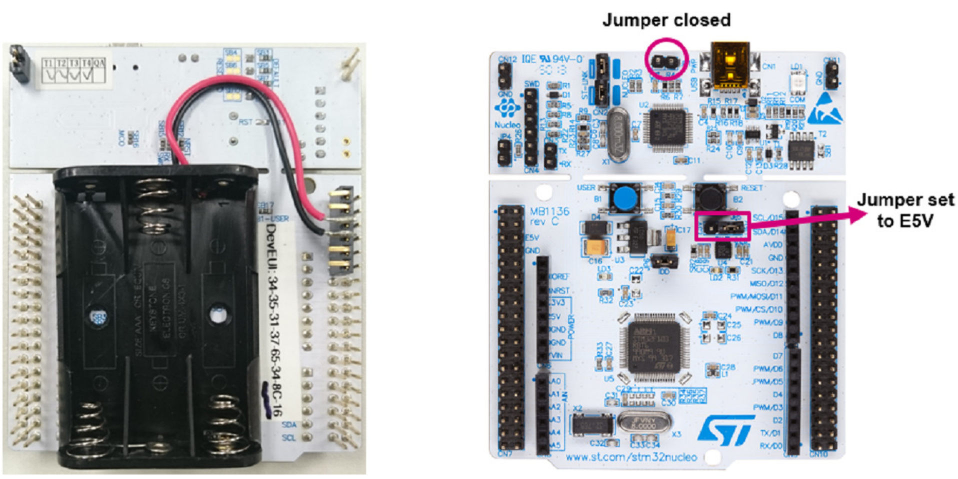
2.1 NUCLEO-L073RZ board

Information about the STM32 Nucleo board is available on *www.st.com.*

The NUCLEO-L073RZ board embeds an STM32L073RZ MCU, a 32-bit microcontroller based on Cortex®-M0+ with 192-Kbyte Flash memory, 20-Kbyte SRAM: these characteristics, together with its peripheral set, enable the LoRa middleware stack (I-CUBE-LRWAN) to run. Moreover, the NUCLEO-L073RZ board embeds the ST morpho extension pin headers for full access to all STM32 I/Os and an on-board ST-LINK/V2-1 debugger/programmer with SWD connector able also to manage the serial communication with the STM32L073RZ.

User can power up the NUCLEO-L073RZ board by connecting it to a PC through a USB cable. However, the ideal setup for the end-device is to be battery operated. Attach a battery holder for 3 x Alkaline AAA (or AA) batteries at the back of the NUCLEO-L073RZ board and connect/solder the +/- terminals to the power pins on the ST morpho connector as shown in *Figure 3*. Set the jumper JP5 to E5V so that the board takes the power from the battery. Now the resistor R32 can be removed to reduce the current consumption of the board. Put a jumper on JP1 to allow the ST-LINK debugger to release the reset pin of the target STM32, when the ST-LINK USB is not connected and enumerated on the PC. This allows the STM32 to execute the firmware with the NUCLEO-L073RZ board connected to the PC through the USB.

Figure 3. NUCLEO-L073RZ board: top view (right) and with batteries (left)

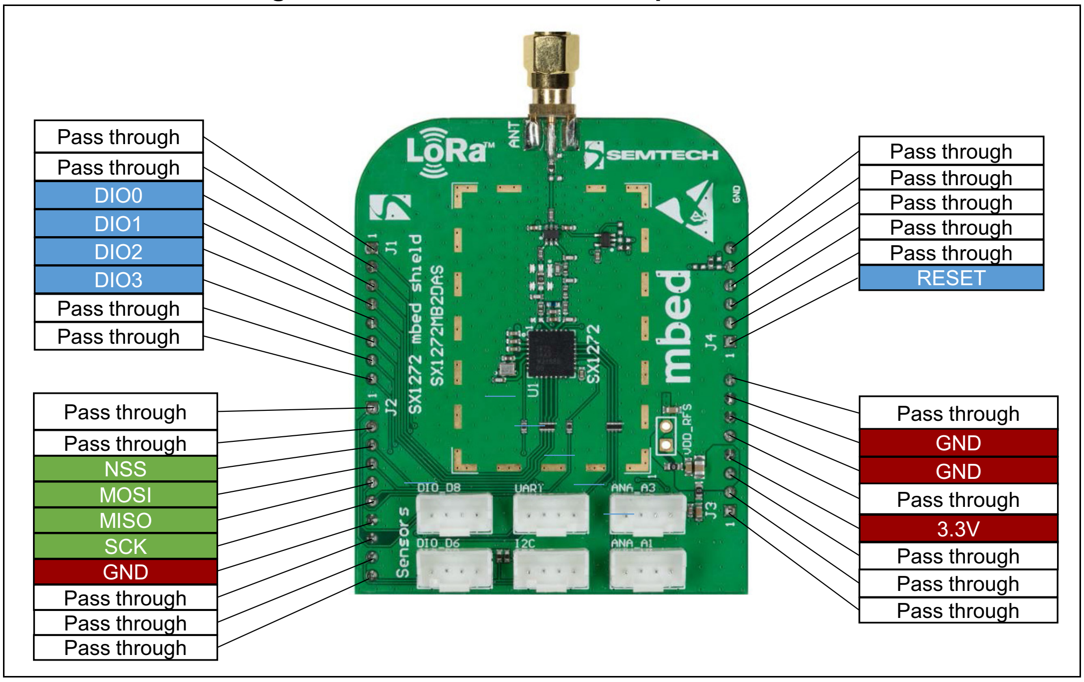


2.2 I-NUCLEO-SX1272D expansion board

I-NUCLEO-SX1272D is the STMicroelectronics code name of the Arm® Mbed™(a) shield SX1272MB2DAS. It embeds a LoRa RF transceiver SX1272 chip and all associated RF matching and filtering components enabling one-single, 50 Ω antenna port. I-NUCLEO-SX1272D embeds its 32 MHz crystal as well. The I-NUCLEO-SX1272D is fully controlled by SPI and interrupt lines.

Optionally Grove compatible sensors can be plugged on the expansion boards. The *Figure 4* below shows a picture of the board together with the I/O lines.

Figure 4. I-NUCLEO-SX1272D expansion board



2.3 Gateway

2.3.1 Semtech IoT starter kit

This kit is designed to offer a self-contained Plug and Play local loop for all IoT object designers, allowing them to verify that their design is able to connect to a public IoT network, following the LoRaWAN specifications. No connection to an Ethernet network is required,

a. Arm and Mbed are registered trademarks or trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

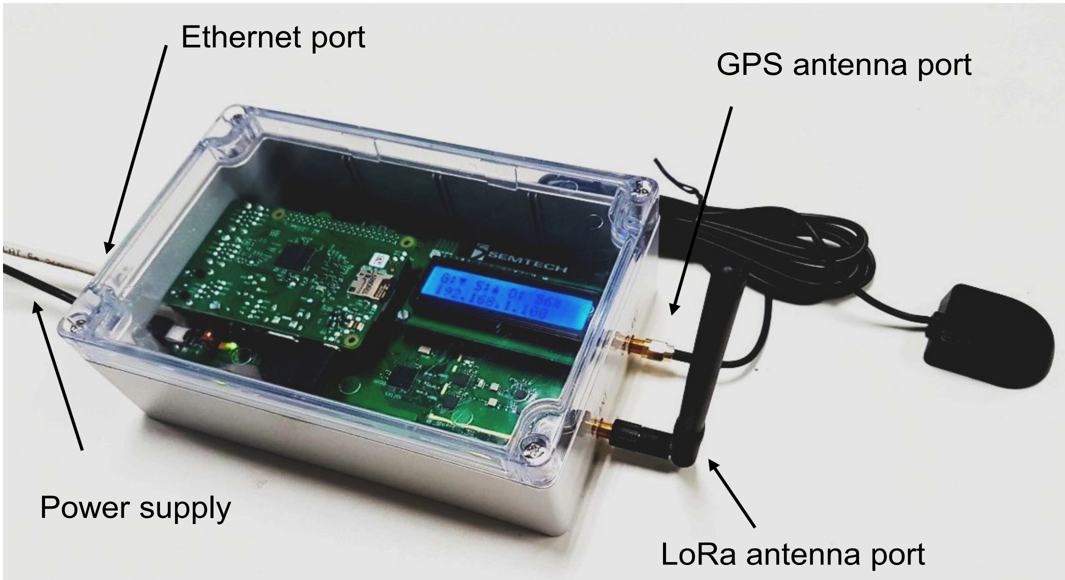
and this simplifies the setup of the starter kit. A simple cross-over Ethernet cable is used to access the kit. It is composed of:

- RaspberryPi B+ with its LoRa IoT shield unit and pre-installed microSD™ card
- SX1301-based concentrator reference design
- Active GPS antenna
- Power adapter (mini USB)

See *Figure 5: Semtech IoT starter kit gateway*.

The Semtech gateway is configured as a local server, meaning that the gateway is performing the network/application server, the web server as well as the gateway functions. No packets are sent to the internet.

Figure 5. Semtech IoT starter kit gateway



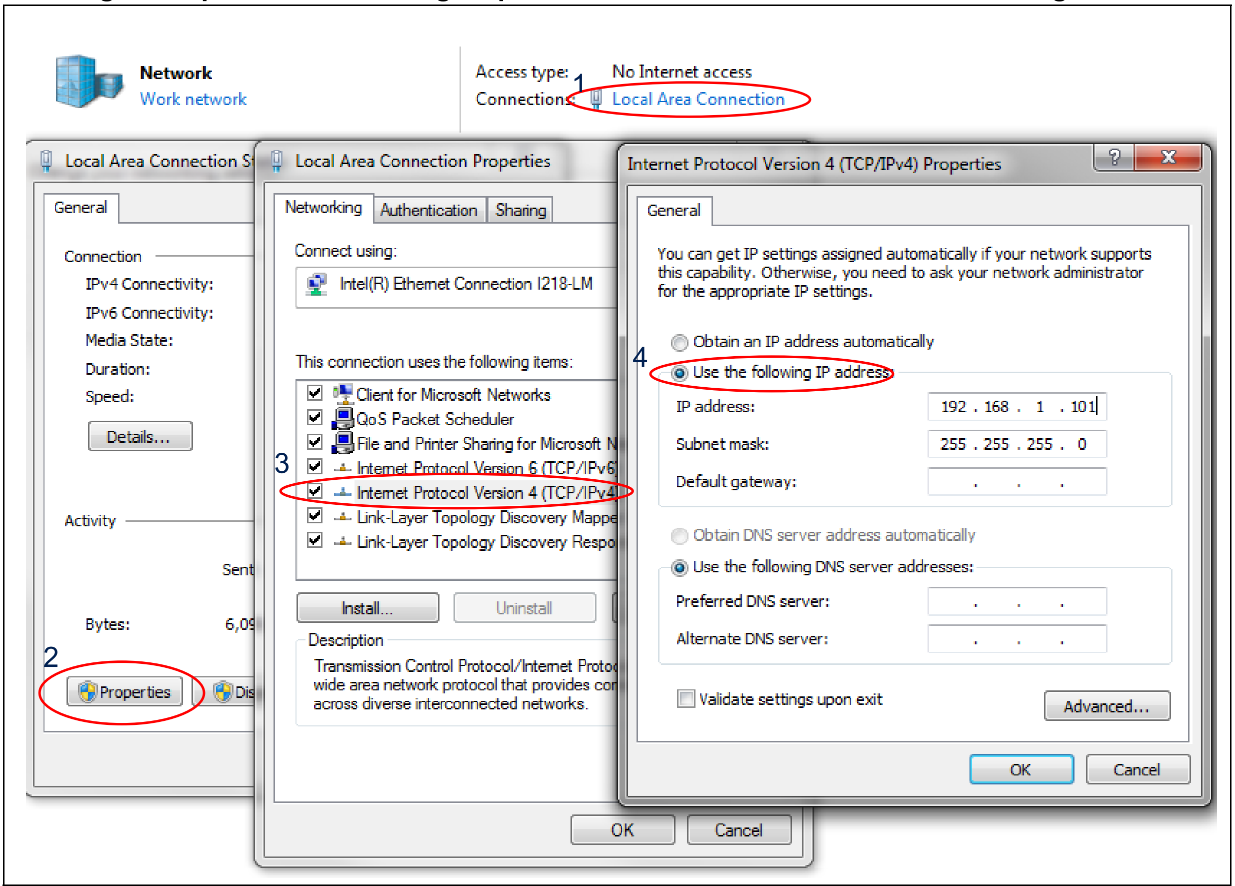
The gateway is powered up using the mini USB connector, the RaspberryPi requires about two seconds to boot. The kit is ready once arrows are upward on the LCD display, as shown below in *Figure 6*:

Figure 6. IP address on LCD display

Two ways are available to get connected to the starter kit:

- Connection to an Ethernet network composed of a router and a DHCP server (the IP address on the LCD screen is then directly entered in the web browser).
- Connection using an Ethernet cable connected between the development PC and the starter kit. In this case the IPv4 address of the development PC must be set to 192.168.1.101, following the steps shown in the *Figure 7*:

Figure 7. Ipv4 address setting sequence in Control Panel/Network and Sharing Center



2.3.2 Other gateways

Kerlink and Multitech provide gateways performing the same function as the starter kit. They also forward the data to a cloud network server and application server through an internal 3G path.

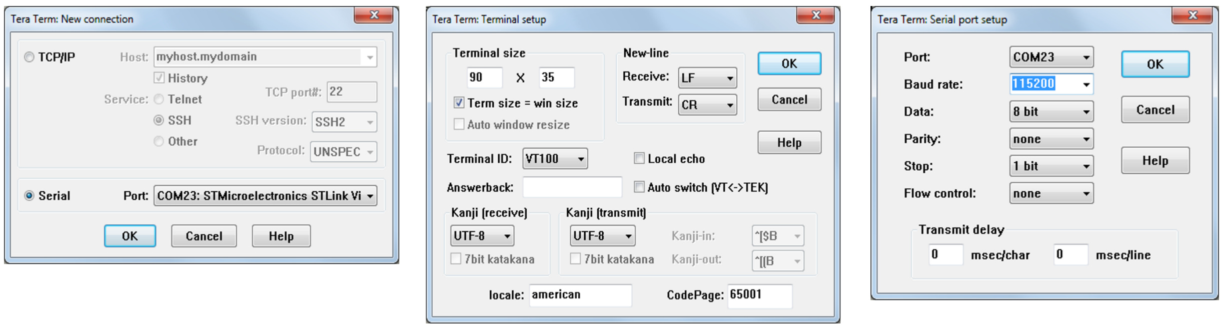
2.4 Installation of the PC tools

Follow the steps below to install the PC tools:

1. Install the preferred Integrated Development Environment (IDE). Three toolchains are supported: IAR™ EWARM, Keil® MDK-ARM™ and AC6 SW4STM32
2. Install a web browser
3. Install the ST-LINK/V2-1 driver
4. Establish the connection with the NUCLEO-L073RZ board by connecting the USB port of the NUCLEO-L073RZ to the USB port of the PC. Allow the PC to enumerate and install the ST-LINK USB drivers
5. Install a terminal software like Tera Term

Connect the NUCLEO-L073RZ board to the PC and open the Tera term. Select the ST-LINK virtual COM port and configure the Tera Term as shown in *Figure 8*:

Figure 8. Terminal setup

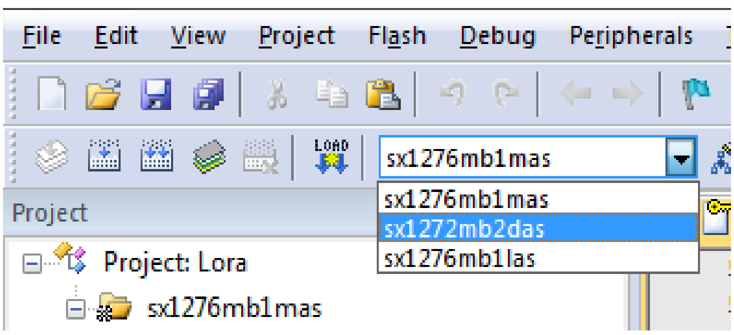


2.5 Firmware setup for the end-device

Follow the steps below to set up the firmware for the end-device:

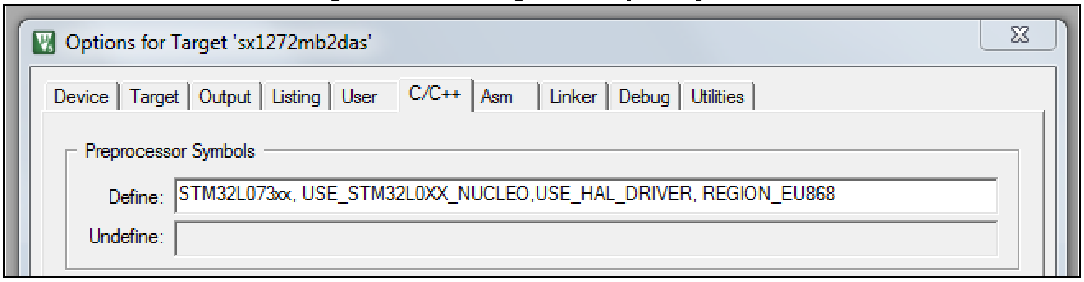
1. Download the latest LoRa firmware package from the www.st.com/i-cube-lrwan webpage and unzip the package to a desired location.
2. In the following figures, Keil IDE is used. Open the Keil project for the "Class A" sample application in \Projects\Multi\Applications\LoRa\classA\MDK-ARM\ STM32L073RZ-Nucleo.
3. Configure the project by selecting the expansion board sx1272mb2das in the drop-down list showed below in *Figure 9*.

Figure 9. Project configuration



4. Define a specific frequency band in the Keil Project Options preprocessor setting. The following bands are supported: REGION_AS923, REGION_AU915, REGION_CN470, REGION_CN779, REGION_EU433, REGION_EU868, REGION_IN865, REGION_KR920, REGION_RU864, REGION_US915, REGION_US915_HYBRID.

Figure 10. Setting the frequency band



2.6 Firmware configuration

Follow the steps below to configure the firmware:

1. Select the activation mode by modifying the definitions in the comissioning.h file as showed in *Table 1*

Table 1. Selection of the activation mode

Defines	Comments
#define OVER_THE_AIR_ACTIVATION	– When set to 1 the application uses the Over-the-Air activation procedure – When set to 0 the application uses the Personalization activation procedure

2. Commission the device (see *Table 2*)

Table 2. Over-the-air activation parameters

Defines	Comments
#define LORAWAN_APPLICATION_EUI	128-bit application EUI
#define LORAWAN_APPLICATION_KEY	128-bit application key
#define STATIC_DEVICE_EUI	– When set to 1 DevEui is LORAWAN_DEVICE_EUI – When set to 0 DevEui is automatically generated using 96-bit STM32 unique ID
#define LORAWAN_DEVICE_EUI	Used when DevEUI auto generation is not enabled.

Example:

```
#define OVER_THE_AIR_ACTIVATION                    1
#define STATIC_DEVICE_EUI                          0
#define LORAWAN_APPLICATION_EUI      { 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,
0x01, 0x01 }
```

```
#define LORAWAN_APPLICATION_KEY      { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE,
0xD2, 0xA6, 0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C }
```

3. Save the modified files and compile the code
4. Program the STM32 by clicking the download button from the toolbar

The following text (*Figure 11*) appears on the terminal after the device starts:

Figure 11. Tera-Term output

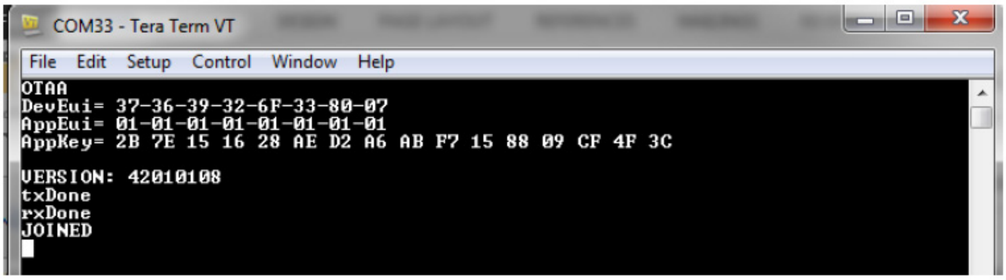


Table 3. Activation by personalization parameters

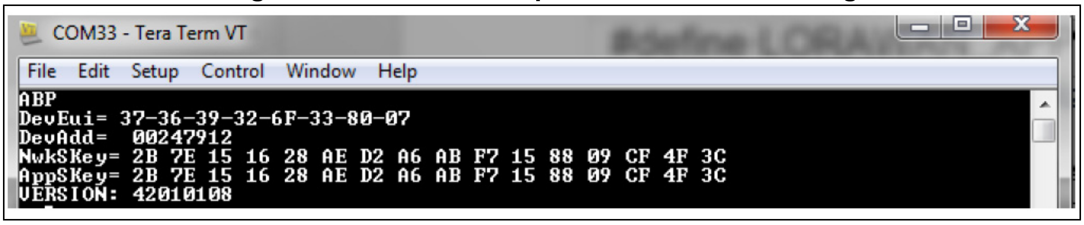
Defines	Comments
#define STATIC_DEVICE_EUI	– When set to 1 DevEui is LORAWAN_DEVICE_EUI – When set to 0 DevEui is automatically generated using 96-bit STM32 unique ID
#define LORAWAN_DEVICE_EUI	Used when DevEUI auto generation is not enabled.
#define LORAWAN_DEVICE_ADDRESS	Used when DevAdd auto generation is not enabled.
#define LORAWAN_NWKSKEY	Used when Activation by Personalization is selected.
#define LORAWAN_APPSKEY	Used when Activation by Personalization is selected.

Example:

```
#define OVER_THE_AIR_ACTIVATION                      0
#define STATIC_DEVICE_ADDRESS                        0
#define LORAWAN_APPLICATION_EUI      { 0xBE, 0x7A, 0x00, 0x00, 0x00, 0x00,
0x00, 0xC8 }
#define LORAWAN_NWKSKEY    { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6,
                             0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C }
#define LORAWAN_APPSKEY    { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6,
                             0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C }
```

The following text (*Figure 12*) appears on the terminal after the device is reloaded and started up using the above parameters:

Figure 12. Tera-Term output after device reloading



5. The Adaptive Data Rate (ADR) feature is enabled in the end-device main.c file (ON by default). When ADR is enabled, the end-device should be static (i.e. not moving)

```
#define LORAWAN_ADR_ON                                  1
```

Note: If an error is encountered during programming, set the debugger to use "Connect under reset" in project options>Debug>ST-LINK Debugger Settings>Debug>Connect & Reset Options.

3 System setup

3.1 Activation parameters for a LoRa end-device

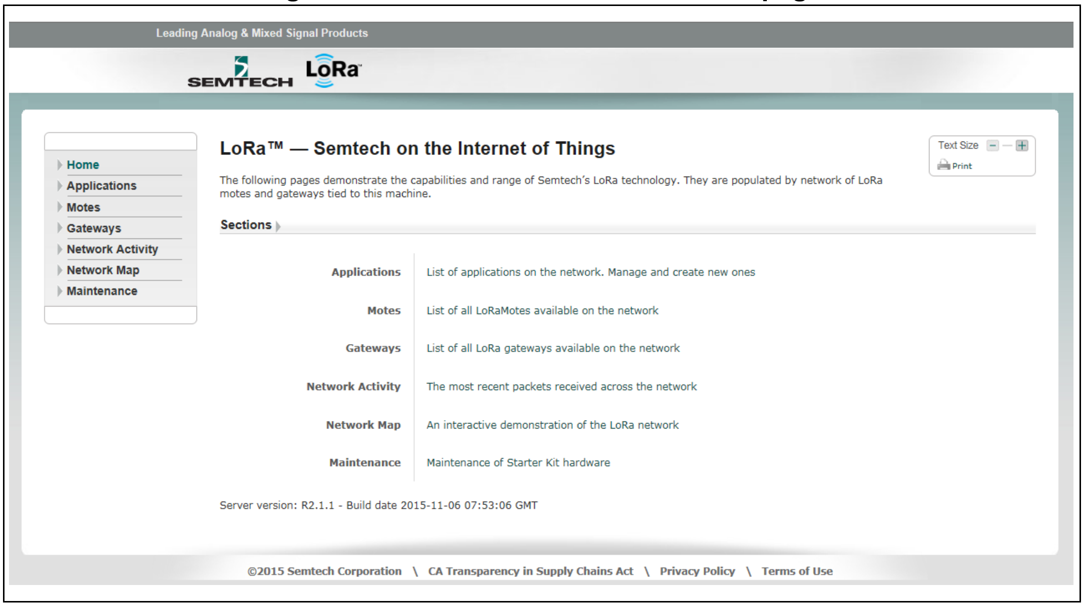
The parameters of the end-device sensor DevEUI, DevAddr, AppEUI, AppKey, AppSKey, and NwkSKey are only shown once after reset. Press the reset button on the NUCLEO-L073RZ board to display them on the terminal (for example the software terminal Tera Term). Take note of them since they are needed when enrolling the end-device to the server.

3.2 Enrolling the end-device to the gateway of the IoT starter kit internal server

To enroll the end-device follow the steps below:

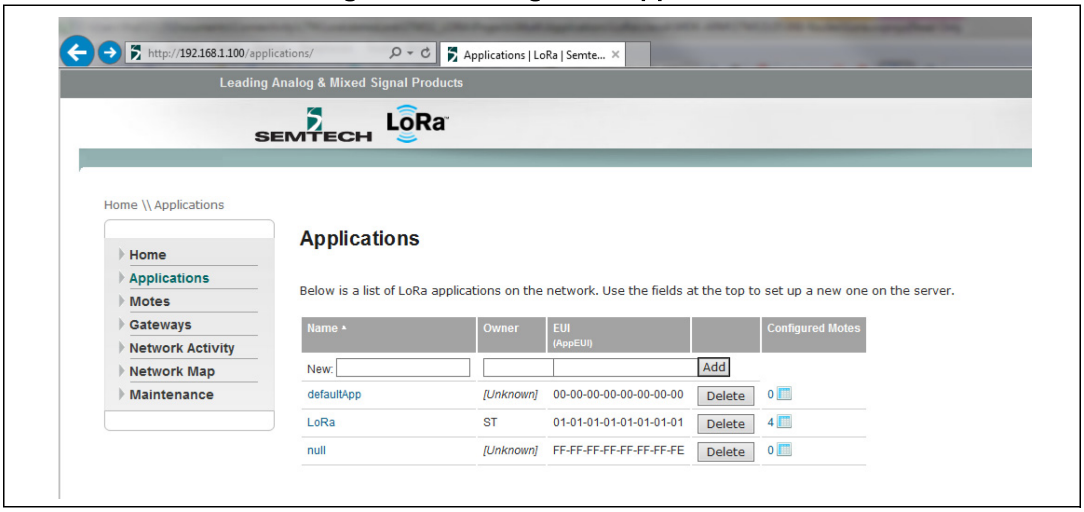
1. Launch a web browser session and enter the gateway IP address into the address bar of the web browser.
2. The following page appears (*Figure 13*):

Figure 13. IoT starter kit internal server page



3. Create a LoRa application. Click on "Applications" and add a new application. The AppEUI flashed in the end-device must be recorded in the gateway, as shown below in *Figure 14*.

Figure 14. Adding LoRa application



4. Add the end-device according to the selected activation mode (see *Figure 15* and *Figure 16*).

Figure 15. Adding OTAA devices

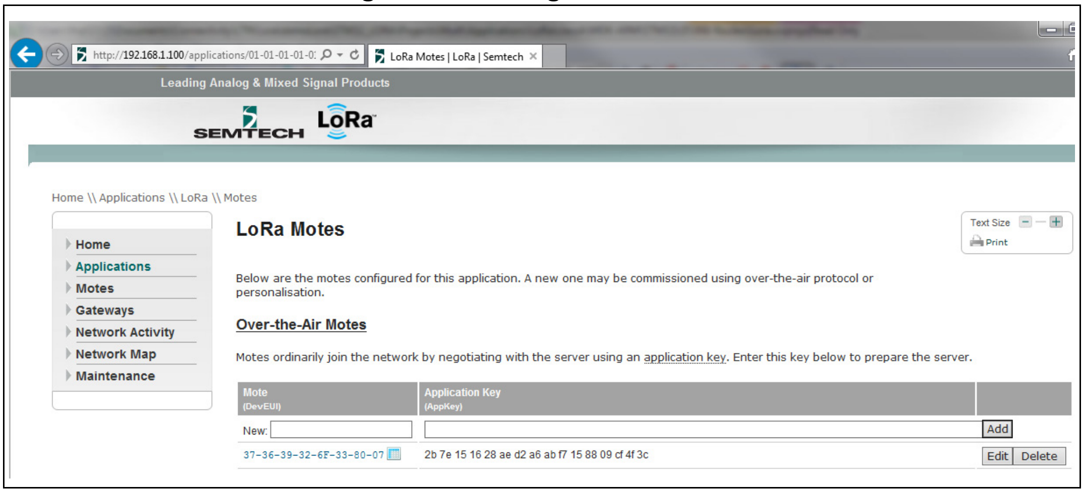
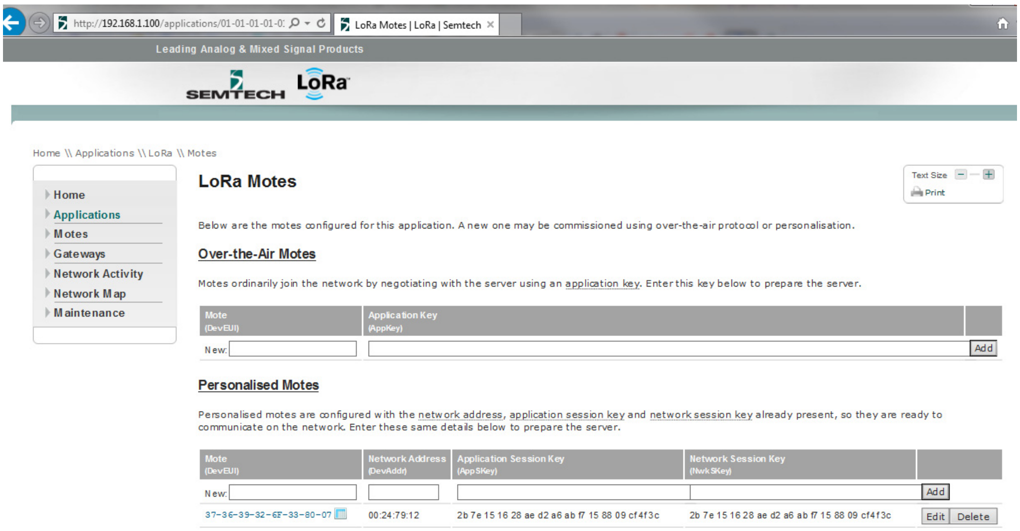
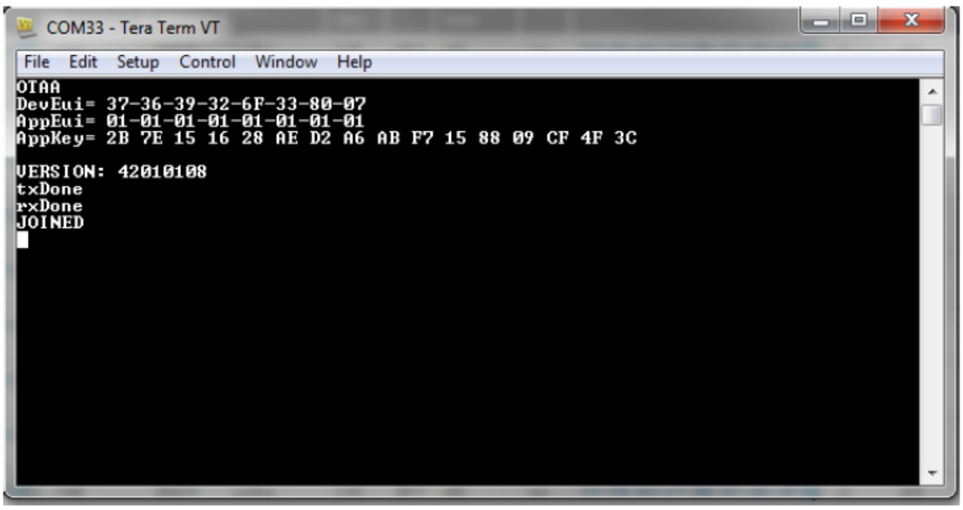


Figure 16. Adding ABP devices



5. Reset the end-device by pressing the reset button on the NUCLEO-L073RZ board. The following text appears on the terminal (*Figure 17*).

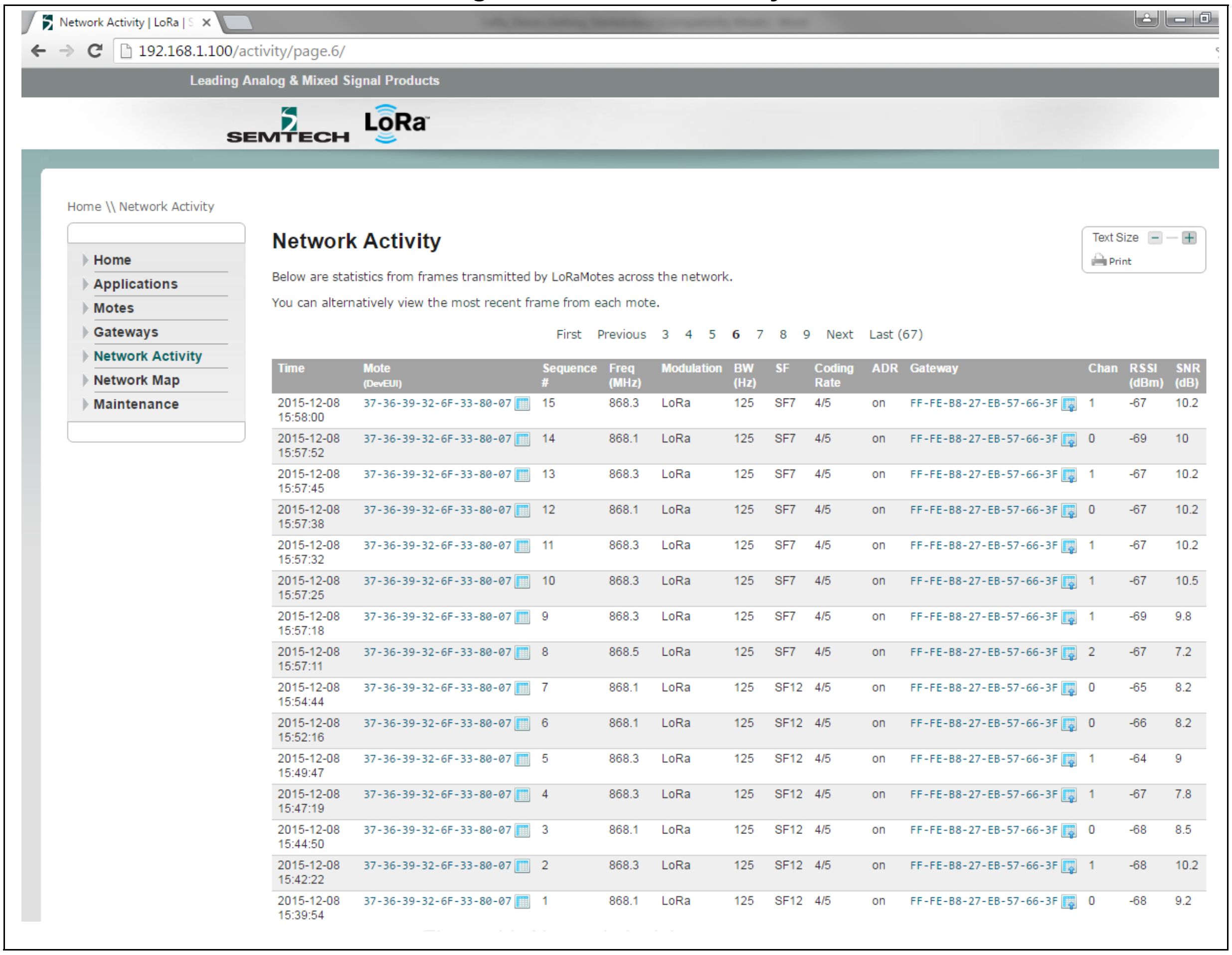
Figure 17. Joined end-device



3.3 Visualization of the IoT starter kit packet

Click on "Network Activity" to view the frames transmitted by the end-device (see *Figure 18*).

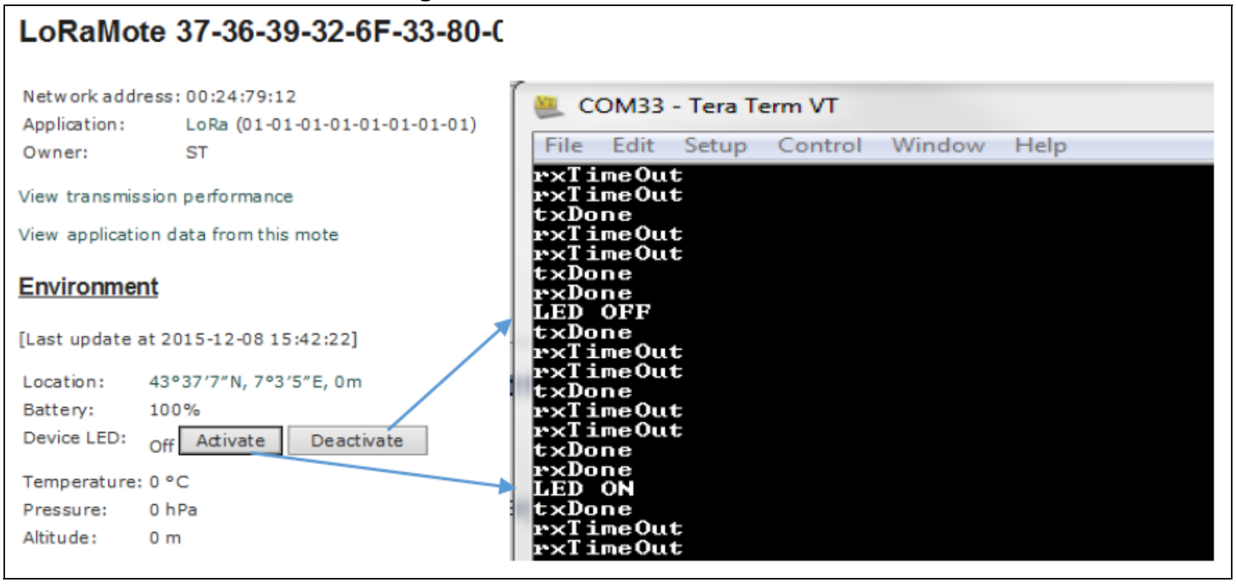
Figure 18. Network activity



3.4 Visualization of the IoT starter kit device data

LED status is changed by clicking on the "Activate" button. A "LED ON" message is printed on the terminal (see *Figure 19*):

Figure 19. Device data visualization

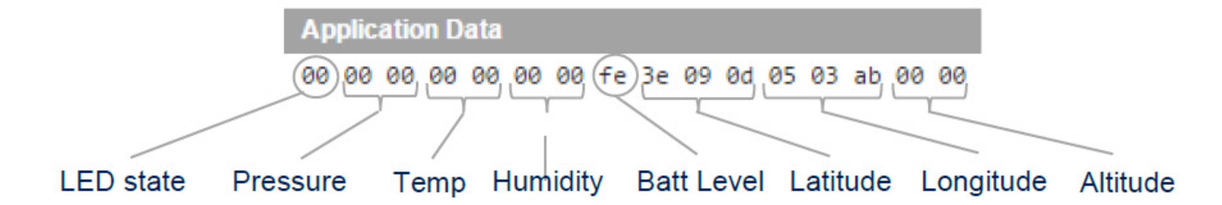


Click on "View application data from this end-device" to view the data payload (see *Figure 20*):

Figure 20. Data payload

2	2015-12-08 16:49:48	473	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:42	472	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:35	471	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:28	470	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:21	469	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:15	468	00 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:08	467	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:49:01	466	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:53	465	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:47	464	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:40	463	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:33	462	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:27	461	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:20	460	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:13	459	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00
2	2015-12-08 16:48:07	458	01 00 00 00 00 00 00 fe 3e 09 0d 05 03 ab 00 00

Figure 21. Example of data payload content



3.5 IoT starter kit shutdown

To prevent a possible corruption of the microSD card files, shutdown the gateway properly before powering off: click on "Maintenance" and click on "Shutdown" (see *Figure 22*).

Figure 22. Shutting down the gateway

4 Extension of the system

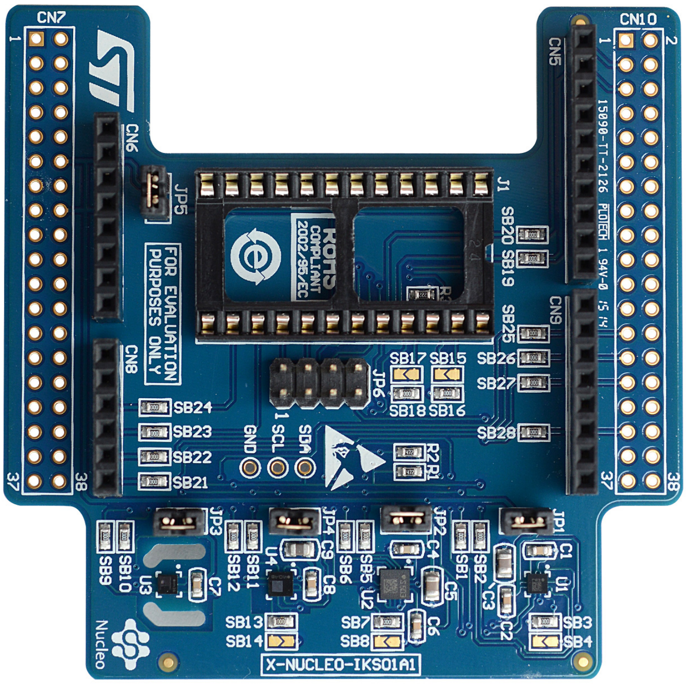
The application presented in *Section 3: System setup* can be enhanced using the X-NUCLEO-IKS01A1 board (see *Figure 23*) in combination with the P-NUCLEO_LRWAN1 pack. The X-NUCLEO-IKS01A1 board hosts four sensors:

- The humidity and temperature sensor (hts221)
- The pressure sensor (lps25H)
- The 3D accelerometer (LSM6DS0)
- The 3D magnetometer (LIS3MDL)

More information are available on the *www.st.com* website.

Only hts221 and lps25H sensors are used in this extension. The sensors are accessed by an I^{2}C interface and the unused interrupts are routed to the STM32.

Figure 23. X-NUCLEO-IKS01A1 board



4.1 Hardware modification

To fit both expansion boards, do the following modifications on the X-NUCLEO-IKS01A1 board:

- Disconnect the I^{2}C lines of the accelerometer and magnetometer (remove sb1, sb2, sb5 and sb6)
- Disconnect the interrupt lines of the accelerometer and magnetometer (remove sb21, sb22 and sb27)
- Remove supply to accelerometer and magnetometer by removing jumpers JP1 and JP2

On SX1272MB2DAS cut DIO3 (J1 Pin 6 must be cut, refer to *Figure 25: I-NUCLEO-SX1272D expansion board*).

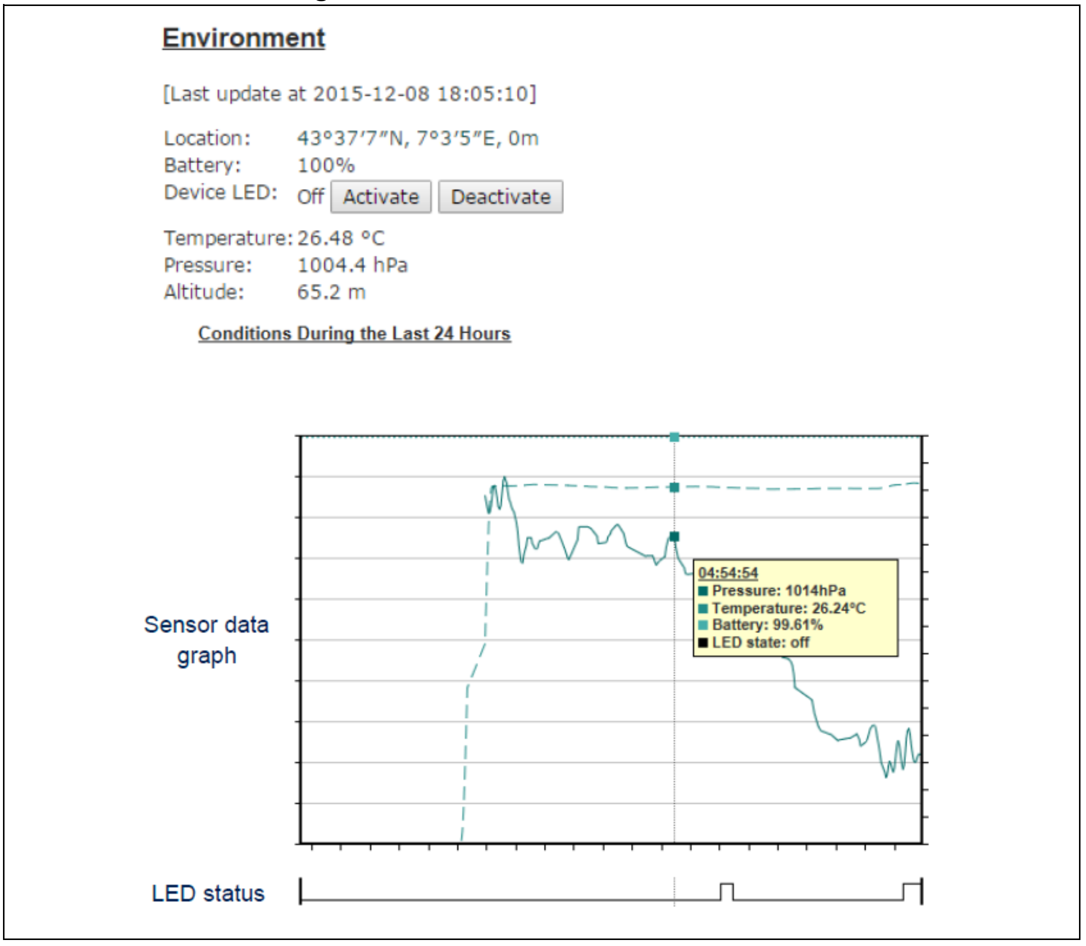
4.2 Firmware modification

When X-NUCLEO-IKS01A1 expansion board is plugged between the NUCLEO-L073RZ board and the SX1272MB2DAS LoRa expansion board, SENSOR_ENABLED must be defined in \Projects\Multi\Applications\LoRa\classA\inc\hw_conf.h.

4.3 IoT starter kit device data visualization

The following *Figure 24* shows the sensor data, sent from the end-device to the LoRa network.

Figure 24. End-device data visualization



5 Ordering information

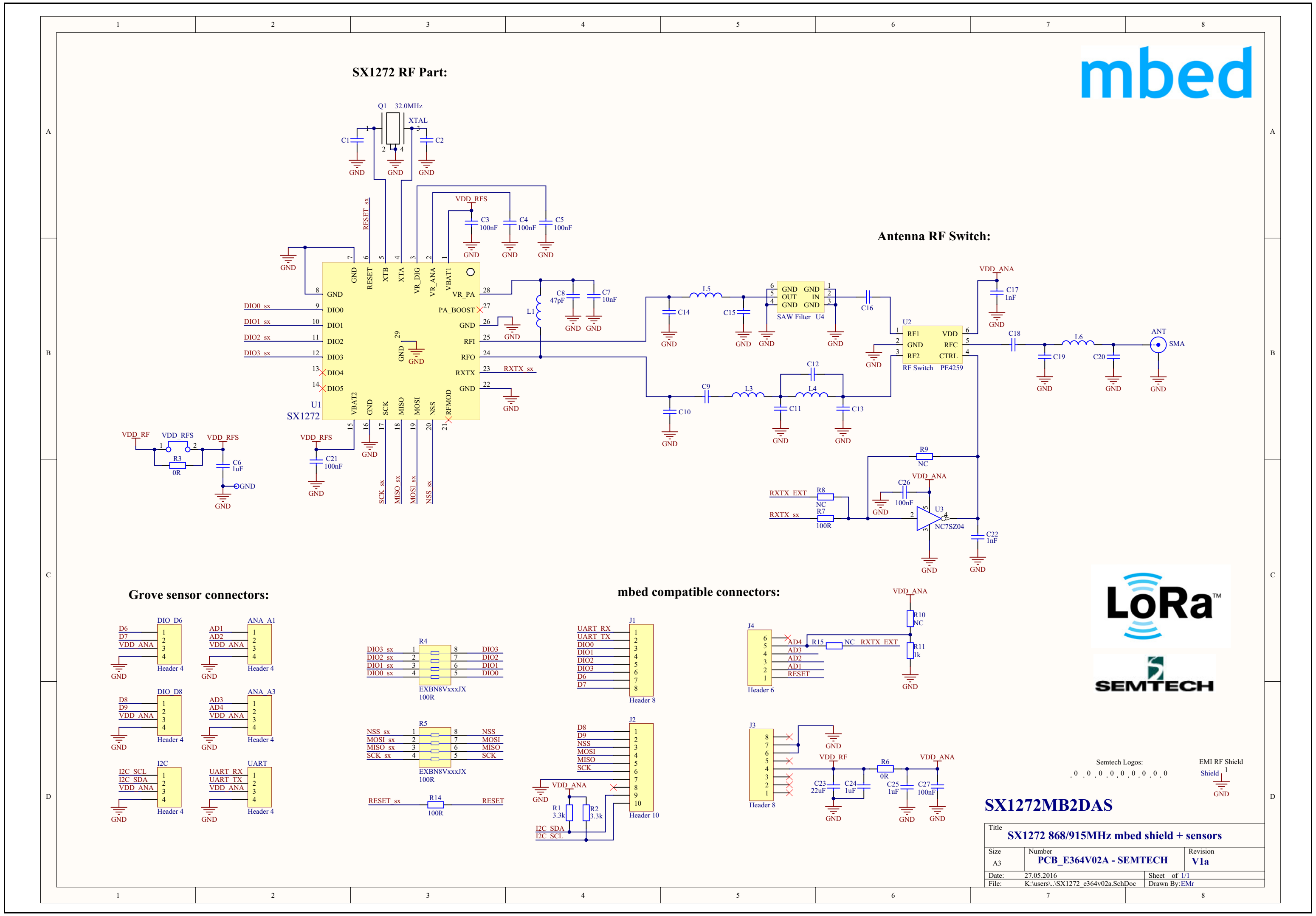
To order the ultra-low-power STM32 and LoRa Nucleo pack, refer to *Table 4*.

Table 4. Ordering information

Order code	Board
P-NUCLEO-LRWAN1	NUCLEO-L073RZ, I-NUCLEO-SX1272D and sub-gigahertz antenna

6 Electrical schematic

Figure 25. I-NUCLEO-SX1272D expansion board



7 Acronyms

Table 5. Acronyms

Acronym	Description
LoRa	Long Range

8 Revision history

Table 6. Document revision history

Date	Revision	Changes
12-Sep-2016	1	Initial version.
18-Nov-2016	2	Updated *Figure 8: Terminal setup*.
9-Jul-2018	3	Updated *Introduction*, *Section 2.5: Firmware setup for the end-device* and *Figure 10: Setting the frequency band*