UM2084
User manual

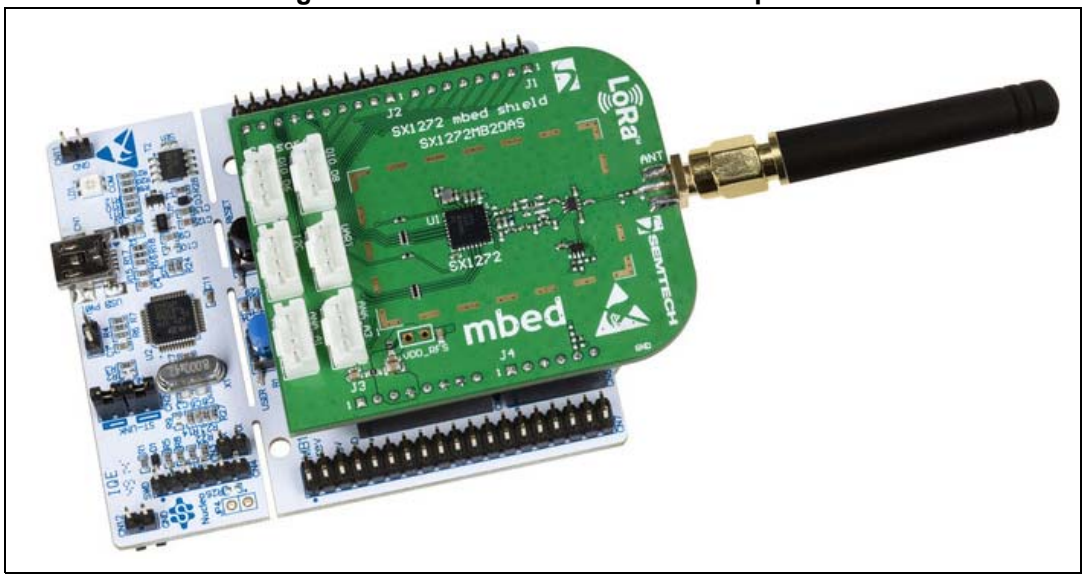
Getting started with the ultra-low-power STM32 and LoRa™ Nucleo pack

Introduction

The ultra-low-power STM32 and LoRa™ Nucleo pack (P-NUCLEO-LRWAN1) is a kit based on a NUCLEO-L073RZ board and on a SX1272MB2DAS LoRa™ RF expansion board from Semtech corporation. The expansion board includes the low-power transceiver SX1272 which features the long-range modem LoRa™. This modem provides high-performance LoRa® modulation as well as OOK / FSK modulation. The P-NUCLEO-LRWAN1 Nucleo pack is compatible with the I-CUBE-LRWAN1 software expansion, a certified middleware stack, compliant with the LoRaWAN™ specifications V1.0.1. It provides support for bidirectional end-devices in Class-A and Class-C protocols and for end-devices activation either through over-the-air activation (OTAA) or activation by personalization (ABP).
For all the details refer to *STM32 LoRa™ software expansion for STM32Cube* databrief (DB2961) at the *www.st.com* website.

This document describes the hardware environment required to build the system and to run applications based on the P-NUCLEO-LRWAN1 Nucleo pack.

Figure 1. P-NUCLEO-LRWAN1 Nucleo pack



1. Picture is not contractual.

Contents

List of figures

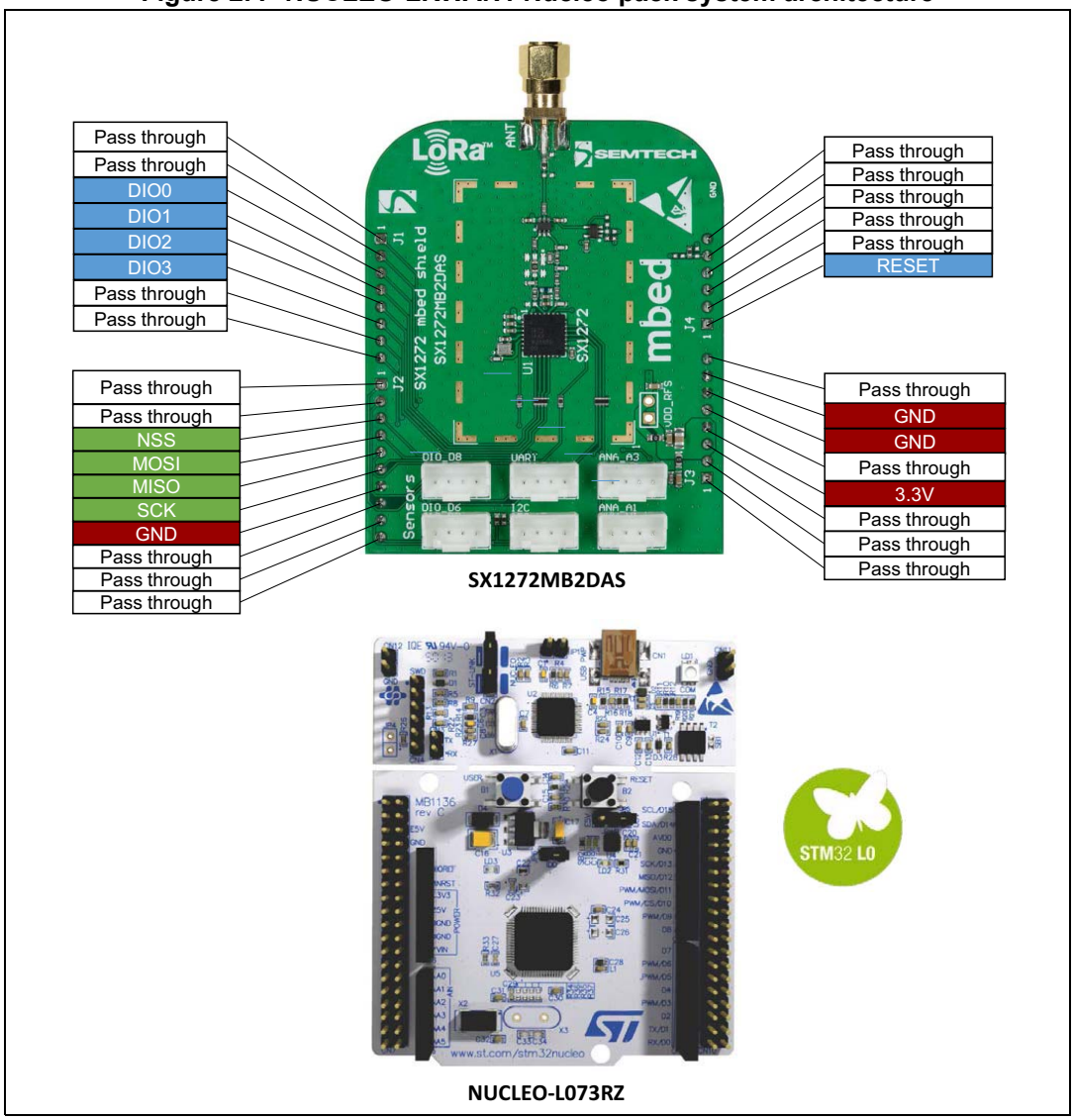
1 Getting started

1.1 System architecture

The ultra-low-power STM32 and LoRa Nucleo pack is composed of two boards (see *Figure 2*):

- The NUCLEO-L073RZ board where the stack runs
- The SX1272MB2DAS LoRa expansion board from Semtech corporation, where a low-power RF transceiver including a LoRa modem, is mounted

Figure 2. P-NUCLEO-LRWAN1 Nucleo pack system architecture



1.2 The P-NUCLEO-LRWAN1 demonstrations

The P-NUCLEO-LRWAN1 Nucleo pack is a complete hardware development platform to prototype solutions based on the LoRa technology with a STM32.

The middleware stack (I-CUBE-LRWAN) offers a set of dedicated APIs and configuration templates to build end-devices, called LoRa objects. LoRa objects are able to join a LoRa network and to communicate through LoRa technology, allowing users to develop easily LoRa applications. For more details refer to the *Ultra-low-power STM32 and LoRa® Nucleo pack with NUCLEO-L073RZ and I-NUCLEO-SX1272D RF expansion boards* User manual (UM2085) at the *www.st.com* website.

Two firmware examples for the STM32L073RZT6 of the NUCLEO-L073RZ board are provided:

- Class A: is a simple demonstration of a LoRa object ready to connect to a LoRa gateway and join a LoRa network (see *Section 1.3.1: Class A demonstration*).
- PingPong: features a communication example between two end-devices when a LoRa gateway is not present (see *Section 1.3.2: PingPong demonstration*).

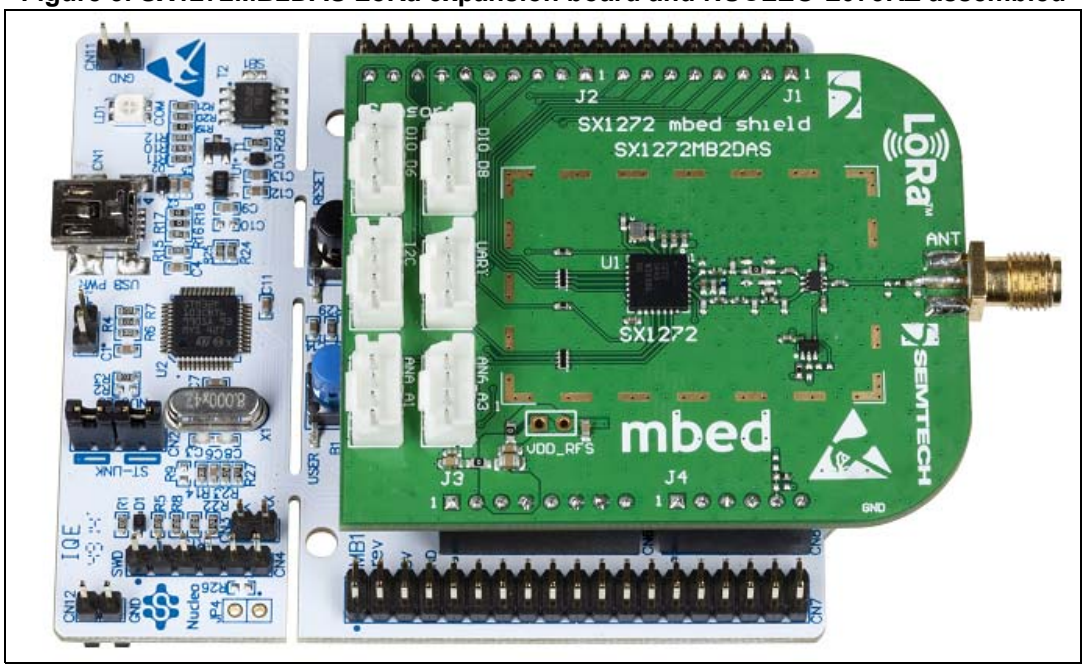
The user can develop other demonstrations with additional hardware, as explained in *Section 1.3.3: Another demonstration with additional hardware*.

1.3 Configuration setup

To run the embedded demonstrations, follow the hardware configuration steps shown below:

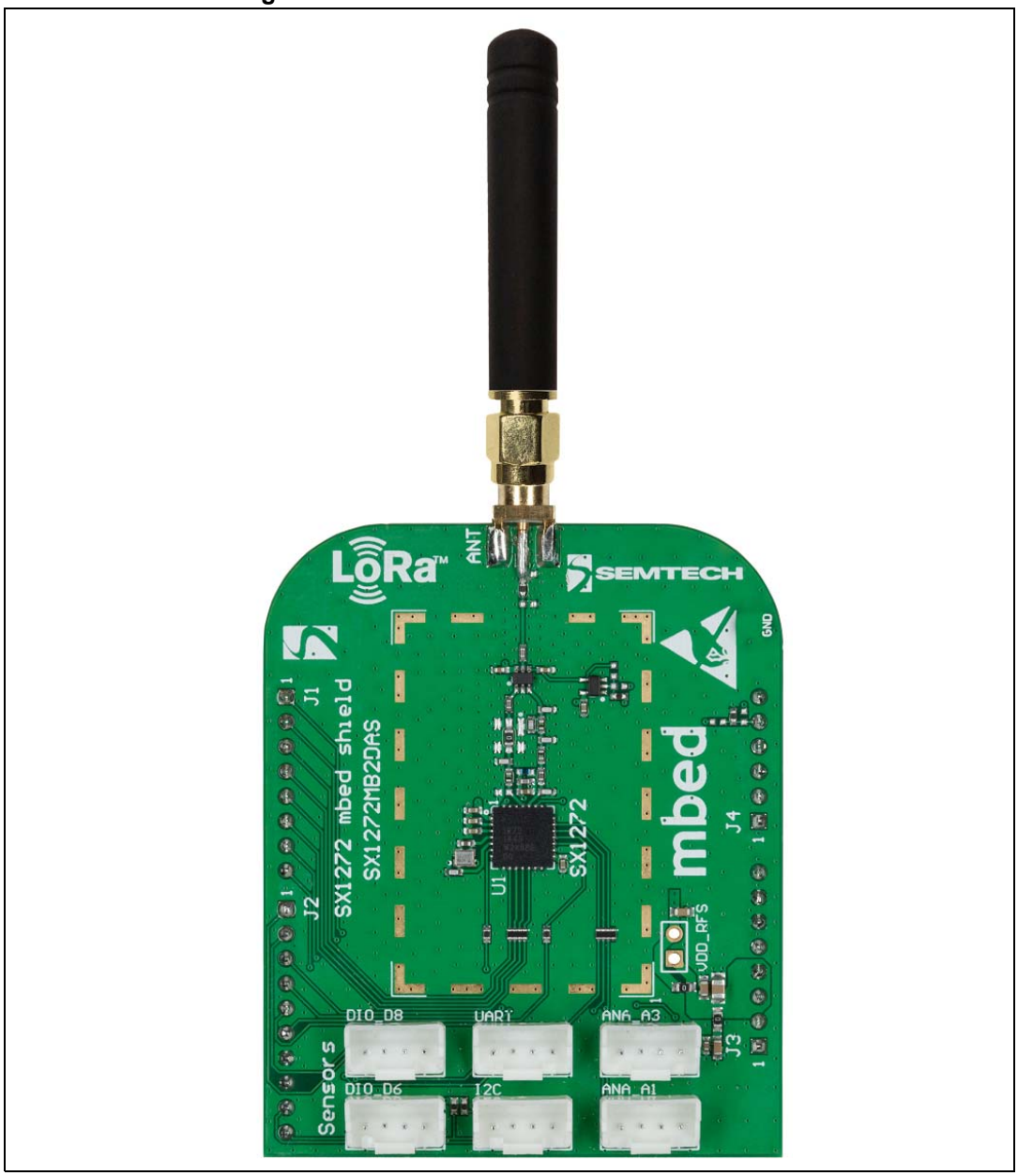
1. The SX1272MB2DAS board must be stacked on a NUCLEO-L073RZ board through the Arduino™ Uno V3 connectors. Only one position is possible for this connection, since the number of pins of the Arduino connectors on each side is different (see *Figure 3*).

Figure 3. SX1272MB2DAS LoRa expansion board and NUCLEO-L073RZ assembled



2. On the NUCLEO-L073RZ board, verify the jumper settings: JP1 must be opened, JP5 (PWR) must be closed on U5V and JP6 must be closed (IDD).
3. On the SX1272MB2DAS expansion board, it is recommended for a better performance - but not mandatory - to screw an antenna on the ANT connector. Refer to *Figure 4*.
4. Connect the NUCLEO-L073RZ board to a PC with a USB Type-A to Mini-B cable (not provided in the package) to power the system.

Figure 4. SX1272MB2DAS board with antenna



1.3.1 Class A demonstration

To run this example, load the Class-A firmware from the I-CUBE-LRWAN STM32Cube software expansion inside the STM32L073RZT6. Check that the “connect under reset” option is set in the debugger settings. As soon as the board is powered up through its USB connector, the LoRa Nucleo pack starts emitting LoRa packets and tries to join a network either through over-the-air activation (OTAA) or activation by personalization (ABP). The activation mode depends on the configuration selected in the comissioning.h header file.

The LoRa Nucleo pack also outputs information on a serial COM port (RX pin of the CN3 connector and GND pin of the CN4 connector) configured as follows:

- Baud rate: 115200 bauds
- Data: 8 bit
- Parity: none
- Stop: 1 bit
- Flow control: none

When the LoRa Nucleo pack starts to boot, it displays its activation-mode identifier (OTAA or ABP) and the firmware version, as shown in the example below for an OTAA output:

```
OTAA
DevEui= 01-01-01-01-01-01-01-01
AppEui= 01-01-01-01-01-01-01-01
AppKey= 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
VERSION: 43010100
```

The example below shows the information displayed for an ABP output:

```
ABP
DevEui= 01-01-01-01-01-01-01-01
DevAdd= 0100000A
NwkSKey= 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
AppSKey= 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
VERSION: 43010100
```

Activate trace and/or debug mode inside the hw_conf.h header file, to display more information.

1.3.2 PingPong demonstration

This example requires two P-NUCLEO-LRWAN1 Nucleo packs, with the PingPong firmware from the I-CUBE-LRWAN STM32Cube software expansion loaded in each STM32L073RZT6. Check that “connect under reset” option is set in the debugger settings. This demonstration consists in establishing a simple Rx/Tx RF link between the two LoRa objects. By default, each LoRa object starts as a master and will transmit a "Ping" message, and then it waits for an answer. The first LoRa object receiving a "Ping" message becomes a slave and sends the master a "Pong" message. The ping-pong game between them starts. Both LoRa Nucleo packs send information about their activity on a serial COM port (RX pin of the CN3 connector and GND pin of the CN4 connector) configured as follows:

- Baud rate: 115200 bauds
- Data: 8 bit
- Parity: none
- Stop: 1 bit
- Flow control: none

The example below shows the output of the two LoRa Nucleo packs:

- on Ping side (master)

```
it's 0:0:3:542 WU@ 0:50:3:542
dz
OnTxDone
txDone
it's 0:0:3:666 WU@ 0:0:4:666
dz
OnRxDone
RssiValue=-111 dBm, SnrValue=15
rxDone
...PING
it's 0:0:3:806 WU@ 0:50:3:806
dz
OnTxDone
txDone
it's 0:0:3:930 WU@ 0:0:4:930
dz
OnRxDone
RssiValue=-111 dBm, SnrValue=16
rxDone
...PING
```

- on Pong side (slave)

```
dz
OnTxDone
txDone
it's 0:0:3:134 WU@ 0:0:4:134
dz
OnRxDone
RssiValue=-112 dBm, SnrValue=14
rxDone
it's 0:0:3:273 WU@ 0:50:3:273
...PONG
dz
OnTxDone
txDone
it's 0:0:3:397 WU@ 0:0:4:397
dz
OnRxDone
RssiValue=-111 dBm, SnrValue=14
rxDone
it's 0:0:3:537 WU@ 0:50:3:537
...PONG
```

RssiValue and SnrValue indicate the quality of the radio link.

1.3.3 Another demonstration with additional hardware

The Class A demonstration can be enhanced with an X-NUCLEO-IKS01A1 board (not provided with the product package). This board featuring various sensors, has Arduino Uno V3 compatible connectors and can be stacked between the NUCLEO-L073RZ board and the SX1272MB2DAS LoRa expansion board from Semtech. The Class-A firmware from the I-CUBE-LRWAN STM32Cube software expansion provides support for three sensors (temperature, humidity and pressure) that are activated with the corresponding compile switch (SENSOR_ENABLED). Once loaded with the Class-A firmware, the P-NUCLEO-LRWAN1 Nucleo pack, equipped with the X-NUCLEO-IKS01A1 board, transmits the values from the three sensors inside the LoRa packets.

2 Revision history

Table 1. Document revision history

Date	Revision	Changes
06-Sep-2016	1	Initial version.
18-Nov-2016	2	Updated baud rate in *Section 1.3.1: Class A demonstration*.