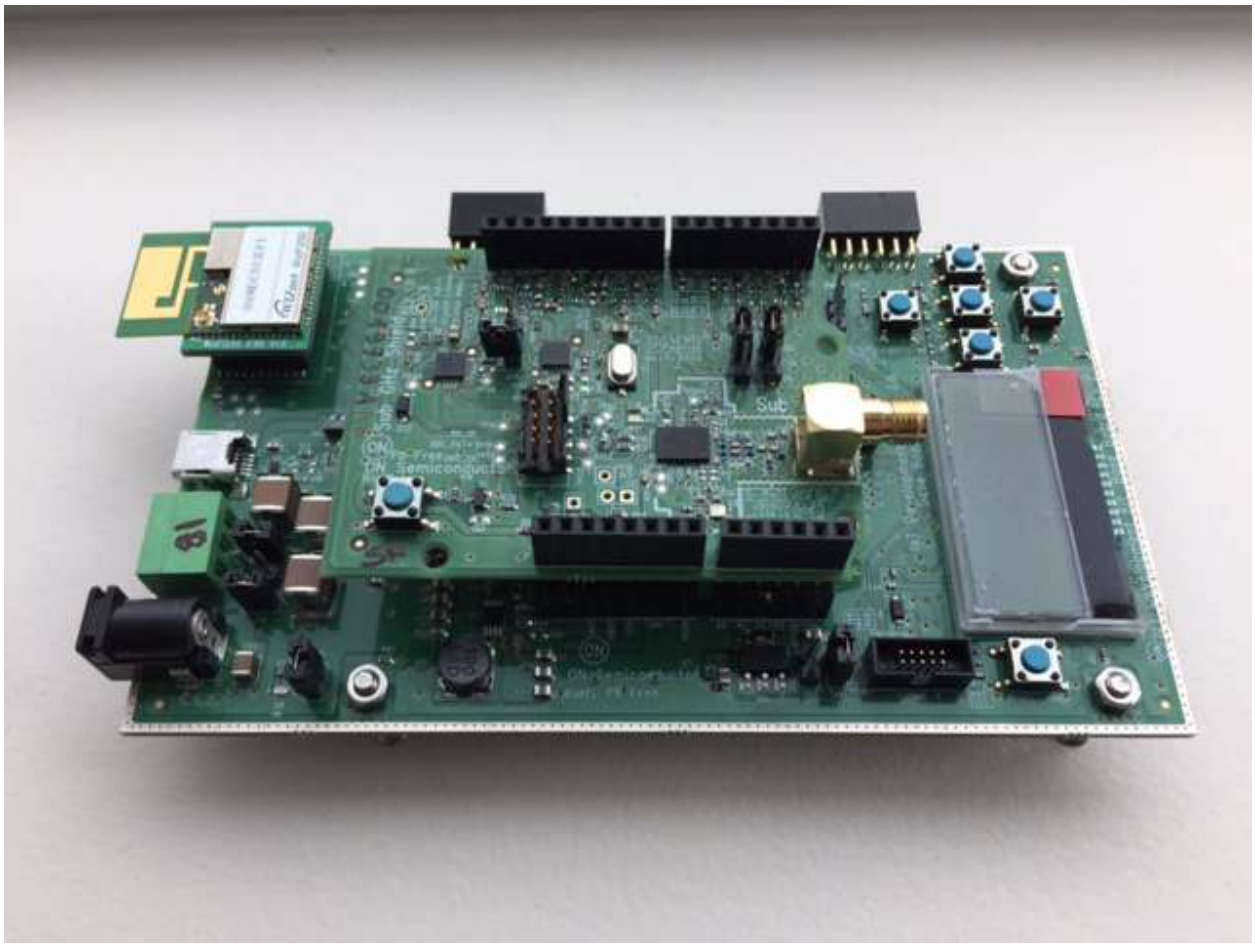
Test Procedure for the SIGFOX-GEVB Evaluation Board

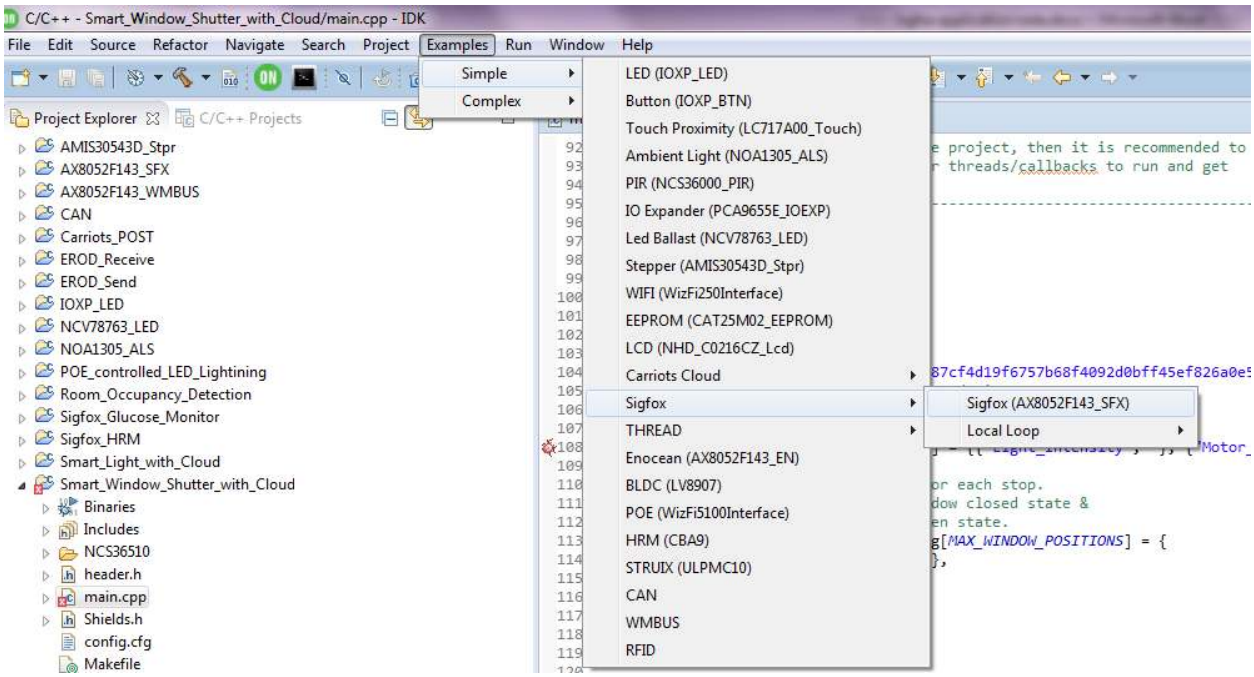
Step 1:

Connect the ON Semiconductor Sigfox shield on top of a ON Semiconductor Base Board.



Step 2:

Follow instruction to get the Sigfox example Software loaded in the IDE:



Step 3:

The firmware should include the read_out for PAC and Device_ID info readout.
(PAC is a 16 digits Hexadecimal number; DEVICE ID is a 8 digits Hexadecimal Number)

```
// GET PAC Info
        sfx.getChipInfo(PAC, sfxBuff, USR_BUFFER_SIZE);
        sprintf(dataBuf, "PAC = %s\r\n", sfxBuff);
        lcd.displayString(dataBuf);
        pc.printf("Sigfox PAC  = %s\r\n", sfxBuff);
        sfx.cleanBuffer(dataBuf, USR_BUFFER_SIZE);
        sfx.cleanBuffer(sfxBuff, USR_BUFFER_SIZE);
        wait(2);
// GET DEVICE_ID Info
                sfx.getChipInfo(DEVICE_ID, sfxBuff, USR_BUFFER_SIZE);
                sprintf(dataBuf, "DEVICE_ID = %s\r\n", sfxBuff);
                lcd.displayString(dataBuf);
                pc.printf("Sigfox DEVICE_ID  = %s\r\n", sfxBuff);
                sfx.cleanBuffer(dataBuf, USR_BUFFER_SIZE);
                sfx.cleanBuffer(sfxBuff, USR_BUFFER_SIZE);
                wait(2);
```

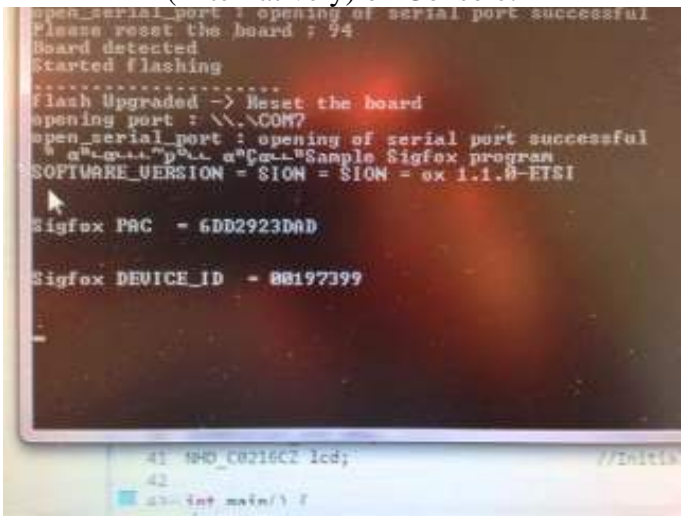
Comment out the following transmission section as your account has not yet been activated:

```
75      //Max number of messages that can be sent to sigfox cloud is 140
76      //This limit of 140 messages is limited by sigfox protocol and not the application
77      //the application or library
78  //  while (count < MAX_SFX_TX) {
79  //      sprintf(dataBuf, "ONSemi %d", count);
80  //      lcd.displayString(dataBuf);
81
82          //param1: Const char data(max of 12 bytes), param2: downlink(1)/no downlink(0)
83          //param3: buffer to contain downlink if expected, param4: size of downlink buffer
84          //Max bytes to be sent is restricted by the AT command of sigfox firmware and not
85          //the library/application
86  //      sfx.sendFrame(dataBuf, 0, sfxBuff, USR_BUFFER_SIZE);
87  //      count++;
88  //      wait(2);
```

Flash the code to your device and enable it so that it will read out the codes:

On LCD:

(Alternatively) on Console:



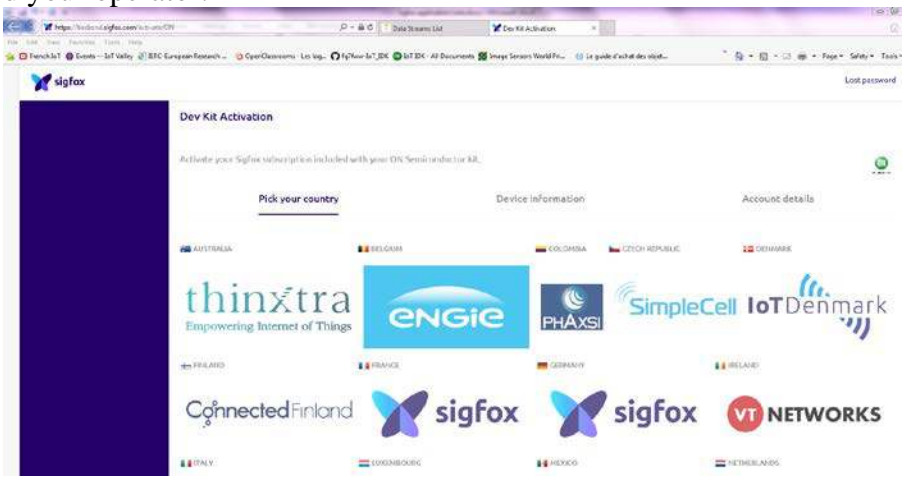
Step 4 (Optional: Should have been completed by default):

Ask your ON Semiconductor marketing contact to get your device activated through Sigfox by providing your representative with the PAC and Device_ID information.

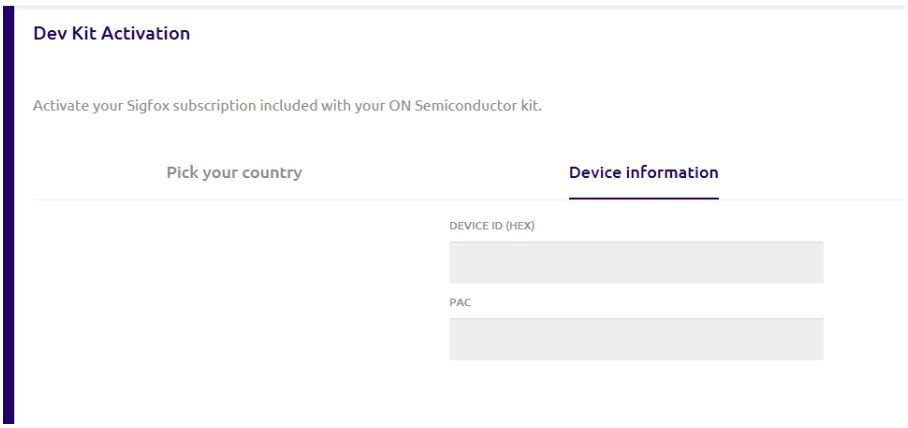
Step 5:

Once activation confirmed by Sigfox (through ON Semiconductor Marketing) create and activate your Sigfox account.
https://backend.sigfox.com/activate/ON

Select country and your operator:



Enter Device ID and PAC info when prompted:



Complete registration information and submit.

A password creation link will be sent to you for next log on.

Step 6:

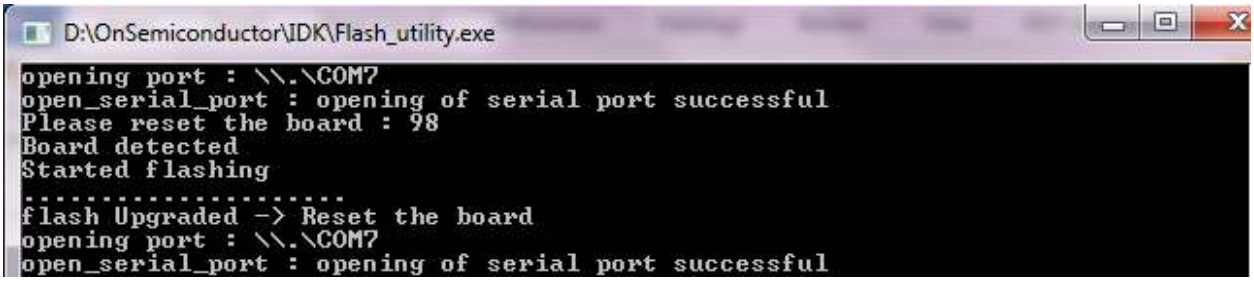
Back to the Firmware and the IoT Kit:
In order to avoid consumption of your daily 140 message maximum (6 Messages per hour)
Modify the code so that messages are only sent twice per reset of the board:

```
#include "mbed.h"
#include "Shields.h"

#define USR_BUFFER_SIZE         128
#define SFX_RET_SUCCESS         0
#define DATA_BUFFER_SIZE        12
#define MAX_SFX_TX              2 // Note : the maximum Frame per days is 140 and the max Frame per hour is 6 .
```

At the send frame section re-establish code section and modify the frame to be sent with “0011223344”

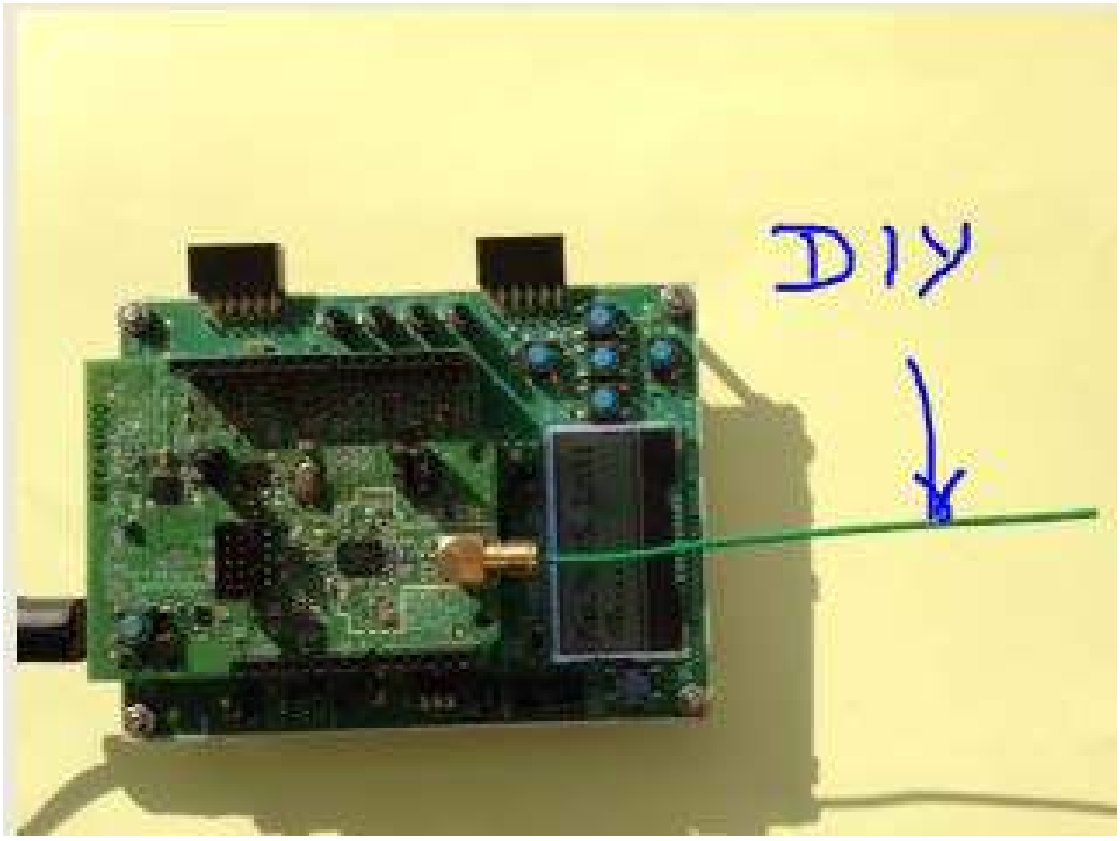
```
    //Max number of messages that can be sent to sigfox cloud is 140
    //This limit of 140 messages is limited by sigfox protocol and not the application
    //the application or library
    while (count < MAX_SFX_TX) {
        sprintf(dataBuf, "ONSemi %d", count);
        lcd.displayString(dataBuf);

        //param1: Const char data(max of 12 bytes), param2: downlink(1)/no downlink(0)
        //param3: buffer to contain downlink if expected, param4: size of downlink buffer
        //Max bytes to be sent is restricted by the AT command of sigfox firmware and not
        //the library/application
        sfx.sendFrame("0011223344", 0, sfxBuff, USR_BUFFER_SIZE);
        count++;
        wait(2);
    }

    lcd.displayString("Max.Msg:2. Exiting.....\r\n");
    pc.printf("Maximum message limit of 2 reached. Exiting.....\r\n");
    return SFX_RET_SUCCESS;
}
```

Flash it to the shield:



Step 7:

Make your DIY ¼ wave antenna:
F= 867MHz => Lambda = 34cm
Cut a piece of electric wire with section similar to the SMA connector central hole.
Wire length is 9cm and remove 5mm of plastic envelope:
This antenna is only 1dB less efficient than of- the-shelf products so it should not affect the connection capability of the kit.

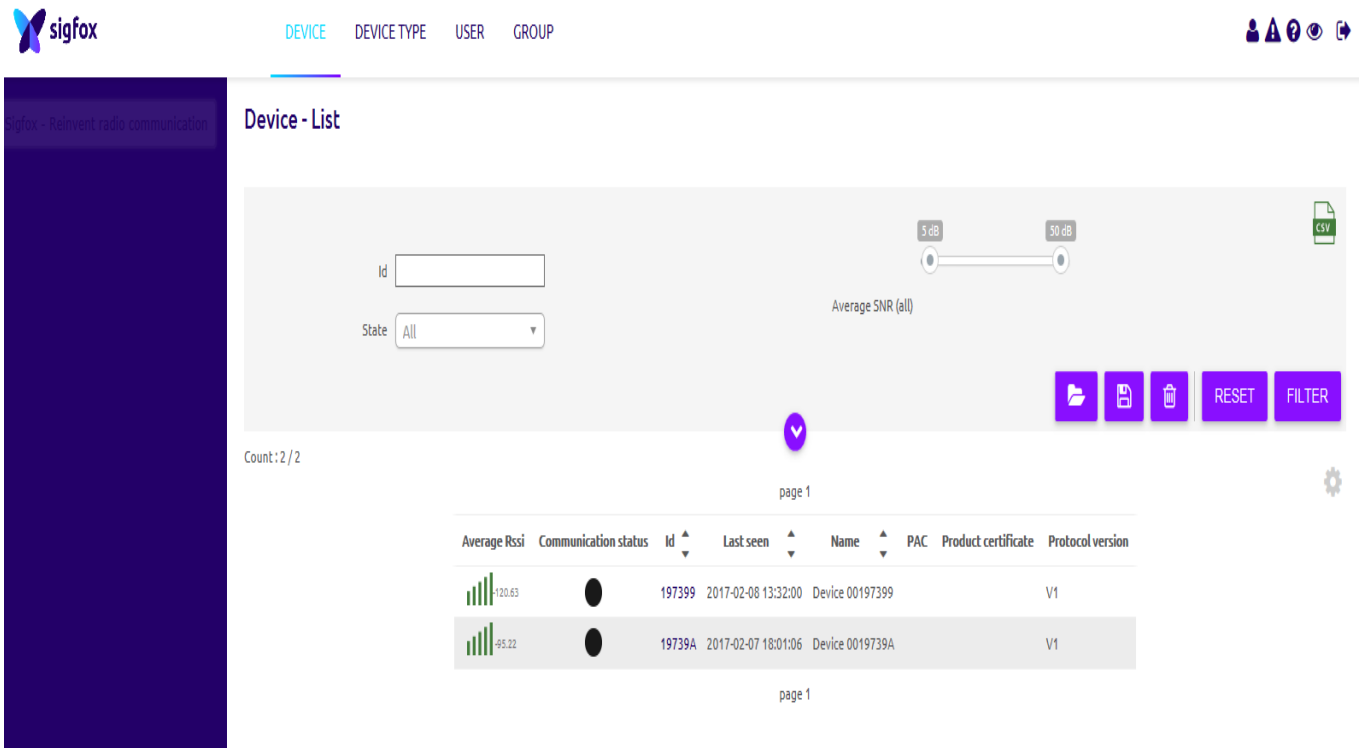


Step 8:

Reset the Board; Communication is started (2 loops)

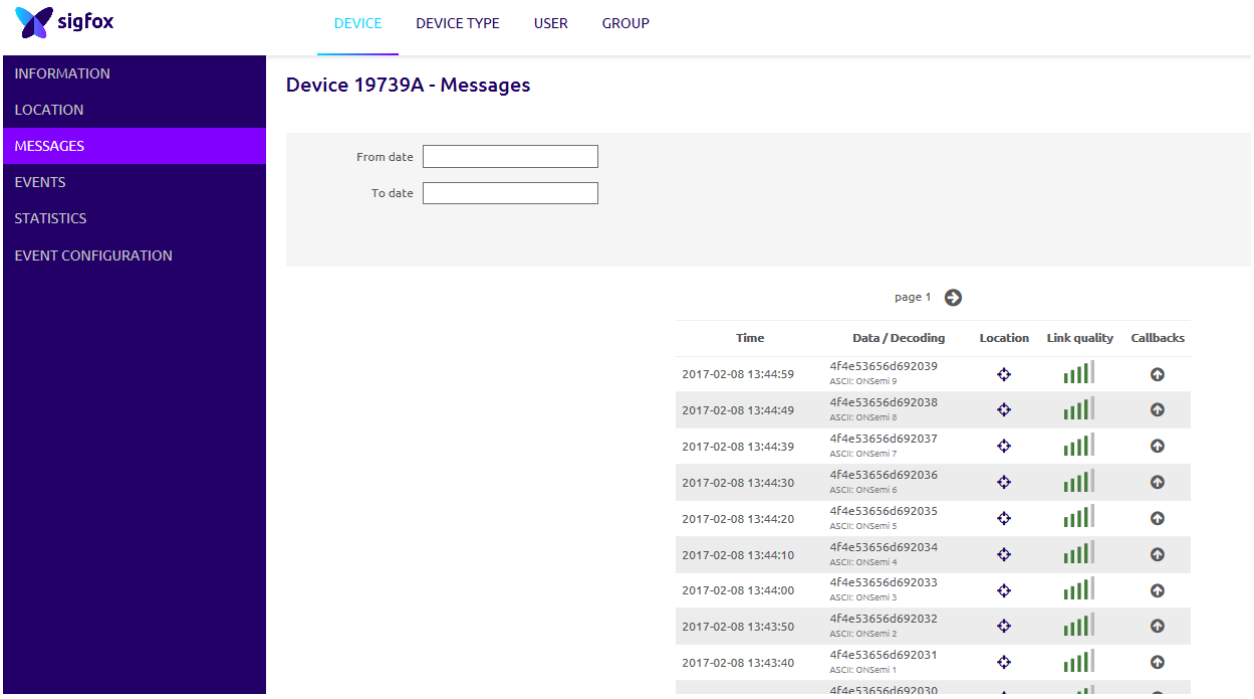
Step 9:

Log to the Sigfox Backend web site: https://backend.sigfox.com/welcome/news
Using the credentials established in step 5.
Go to DEVICE TAB and select your device:



RSSI and SNR perf quasi identical to Commercial antenna show that your device has transmitted message to Sigfox infrastructure; Click on your device ID

Select the MESSAGE tab:



Step 10:

View your message stored on SIGFOX Cloud: