WIRELESS & SENSING PRODUCTS

User Guide to the LoRa® 2.4GHz 3 Channels Single SF Reference Design

Table of Contents

List of Figures

List of Tables

1 Introduction

This user guide introduces the Semtech LoRa® 2.4GHz 3 Channels Single SF Reference Design V1.0 and how to set it up with a Raspberry Pi 3.

The reference design consists on a MCU, four SX1280 RF transceivers, three dedicated to RX and one dedicated to TX, a T/R switch + LNA in a front-end module , and all of the necessary filters and power supplies to deliver a high performance 3 channels single spreading factor LoRa gateway.

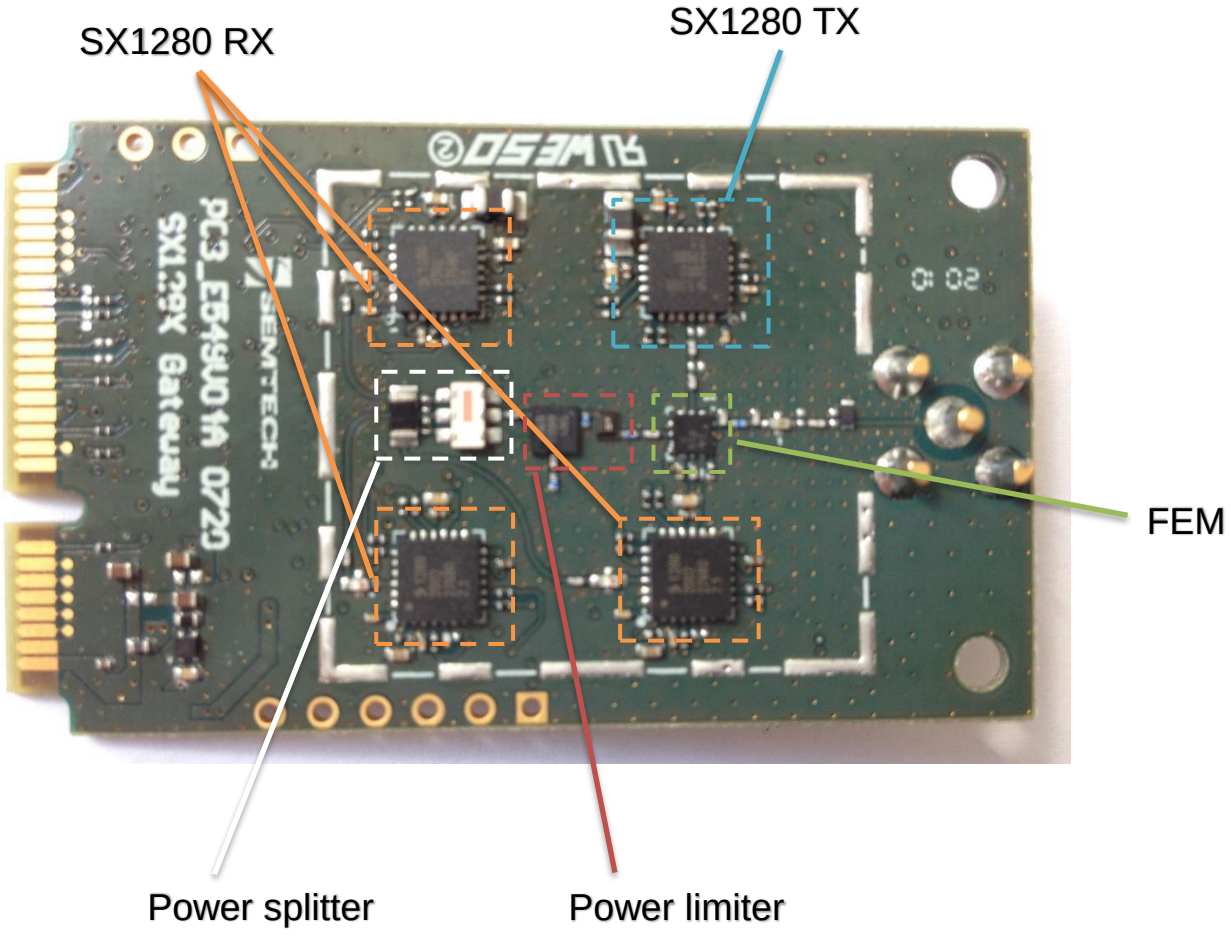

Figure 1: LoRa® 2.4 GHz 3 Channels Single SF Reference Design V1.0 bottom

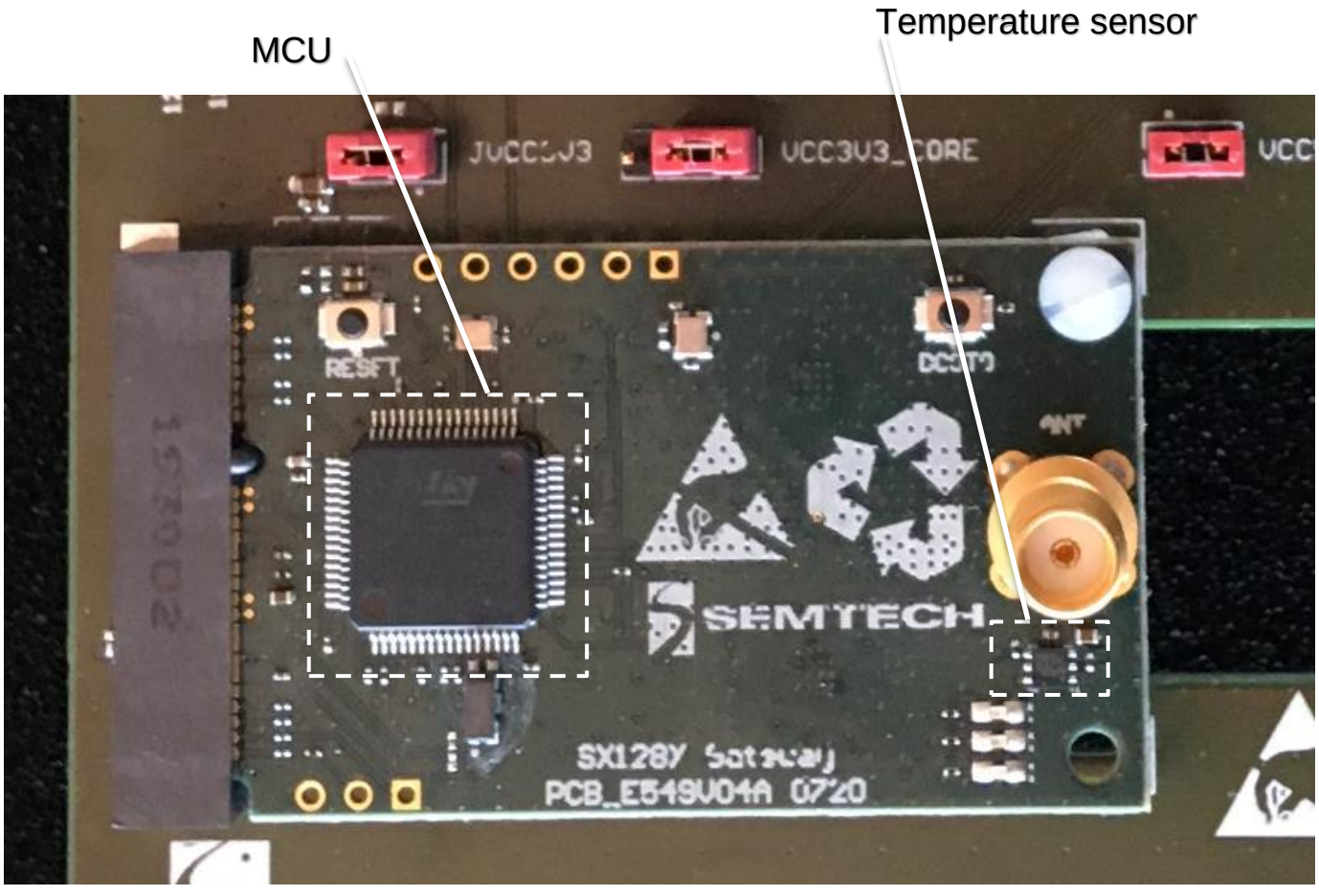

Figure 2: LoRa® 2.4 GHz 3 Channels Single SF Reference Design V1.0 top

2 Hardware Overview

2.1 Absolute Maximum Ratings

Item	Minimum	Typical	Maximum	Unit
Maximum Supply Voltage	-0.3	3.3V	3.6	V
Operating Temperature	-40	25	85	^{o}C
Maximum RF Input Level			+10	dBm

Table 1: Absolute Maximum Ratings

2.2 RF Front-End Architecture

The RF front-end architecture of the 2.4GHz 3 Channels Single SF Reference Design displays the following characteristics:

- Half-duplex mode i.e. can't receive and transmit simultaneously
- Simultaneously receive 3 LoRa® channels single-data rates selectable (SF7 ~ SF12 / 812.5 kHz)
- Maximum transmit output power = +12dBm
- Typical sensitivity level:
    - -116 dBm at SF7 BW 812.5 kHz
    - -129 dBm at SF12 BW 812.5 kHz
- Ability to work in hostile RF environments such as close to cellular mobile phones, WiFi routers, Bluetooth devices

2.3 LoRa® 2.4 GHz 3 Channels Single SF Reference Design block diagram

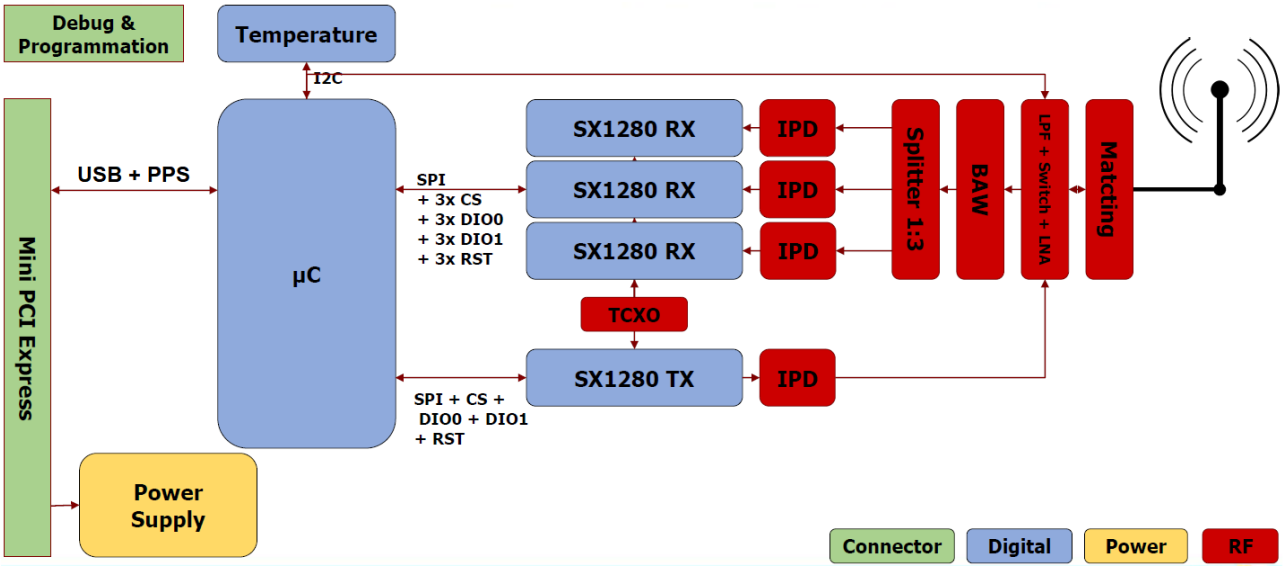

Figure 3: LoRa® 2.4 GHz 3 Channels Single SF Reference Design V1.0 Block Diagram

On-board Mother board main requirements:

- 1 x USB : coming from host to the MCU USB interface
- 3.3V power supply

2.4 Power Consumption

MODE	DESCRIPTION	TYPICAL CURRENT CONSUMPTION	UNIT
IDLE	HAL packet_forwarder OFF	12.6	mA
3 RX CHANNELS ON TX OFF	HAL packet_forwarder ON	33.3	mA
3 RX CHANNELS OFF TX ON AT 13 DBM 2.4 GHZ	HAL packet_forwarder ON	43.5	mA

Table 2: Typical Current Consumption at 5.0 V

3 Software Overview

The 2.4GHz 3 Channels Single SF Reference Design software can be split in two main parts:

The **packet forwarder** is a program running on the host of a LoRa® gateway that forwards RF packets received by the concentrator to a server through an IP/UDP link, and emits RF packets that are sent by the server.

The **gateway_2g4_hal** is a host driver/HAL to build a LoRa® 2.4 GHz 3 Channels Single SF Reference Design which communicates through USB with a concentrator board based on Semtech SX1280 RF transceivers.

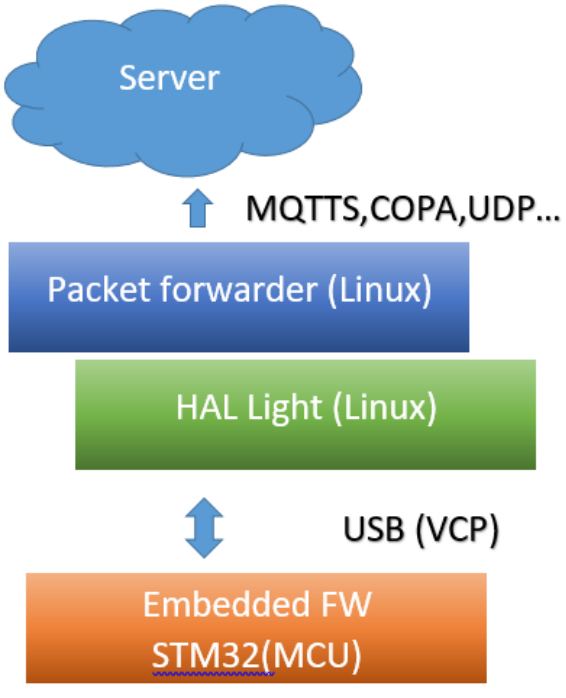

Figure 4: Software Overview

The packet_forwarder (gateway application) as well as gateway_2g4_hal source code can be found under LoRa® Github:

https://ch02git1.semtech.com/lora-2g4-gateway/gateway_2g4_hal

For more details see the readme.md file in the followings directories:

- gateway_2g4_hal
- gateway_2g4_hal/libloragw
- gateway_2g4_hal/libtools
- gateway_2g4_hal/mcu_bin
- gateway_2g4_hal/util_boot
- gateway_2g4_hal/packet_forwarder
- gateway_2g4_hal/util_net_downlink
- gateway_2g4_hal/util_chip_id

For basic testing, utilities such as test_loragw_hal_tx, test_loragw_hal_rx, are provided on the LoRa® Github repository:

https://github.com/Lora-net/ gateway_2g4_hal/libloragw

Notice!
The default configuration file “global_conf.json” is given as an example and may need to be adapted to your design. Several configuration file examples are located in the following directory: [PATH]/ gateway_2g4_hal/packet_forwarder.

4 Use with Raspberry Pi

The Semtech LoRa® Concentrator reference design has been tested with Raspberry Pi 3 model B

https://www.raspberrypi.org/products/

4.1 LoRa® 2.4 GHz 3 Channels Single SF Reference Design + Interface board + Raspberry Pi Connection

Simply connect the 2.4GHz 3 Channels Single SF Reference Design to the interface board through the Raspberry pi USB as depicted on the picture below:

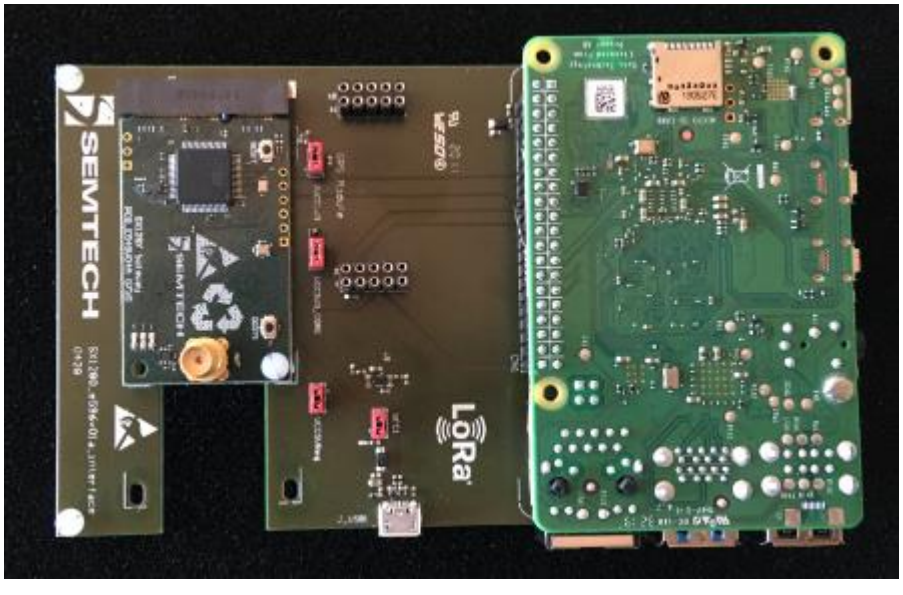

Figure 5: LoRa® 2.4 GHz 3 Channels Single SF Reference Design + interface board + raspberry pi Connection

4.2Raspberry Pi Image Software Installation

- Download the Raspbian image:
  - Go to address https://www.raspberrypi.org/downloads/raspbian/
  - Choose “RASPBIAN BUSTER LITE”
- Refer to following guide to setup your SD card with the downloaded image:

  https://www.raspberrypi.org/documentation/installation/installing-images/
  - Format the SD card:

    https://www.sdcard.org/downloads/formatter/eula_windows/

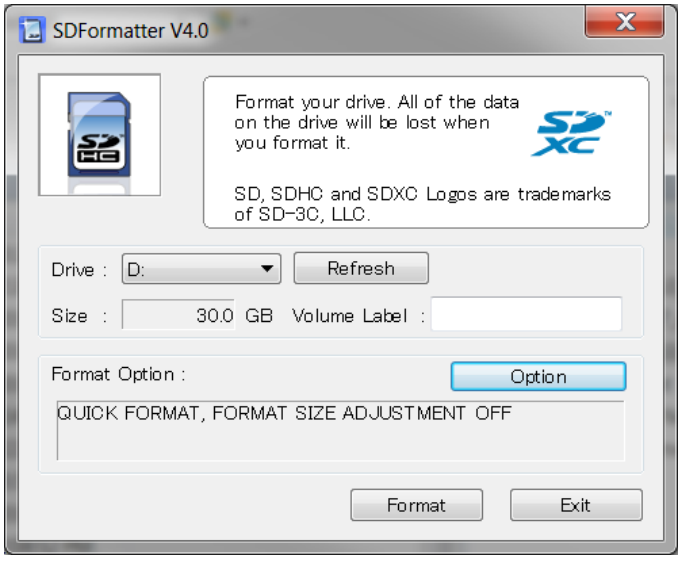

Figure 6: SDFormatter

  - Write the image previously downloaded on the SD card:

    https://sourceforge.net/projects/win32diskimager/

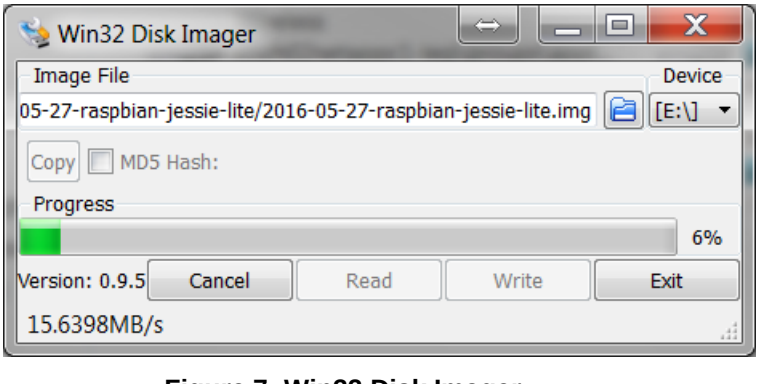

Figure 7: Win32 Disk Imager

4.3Starting Raspberry Pi

Once the SD card is burned, insert it in the Raspberry Pi and choose a way to login Raspberry Pi:

- HDMI monitor and USB keyboard
- SSH connection :
    - Enable SSH by placing a file named “ssh” (without any extension) onto the boot partition of the SD card:

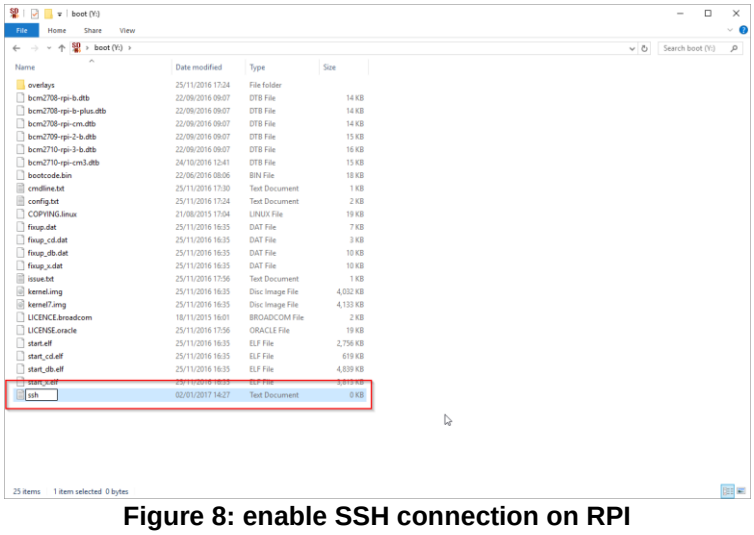

Figure 8: enable SSH connection on RPI

Below is the description through an SSH client enabled from *raspi-config* tool, *Interfacing Option (is activated by HDMI monitor and USB keyboard)*

4.3.1 Login: pi and Password: raspberry

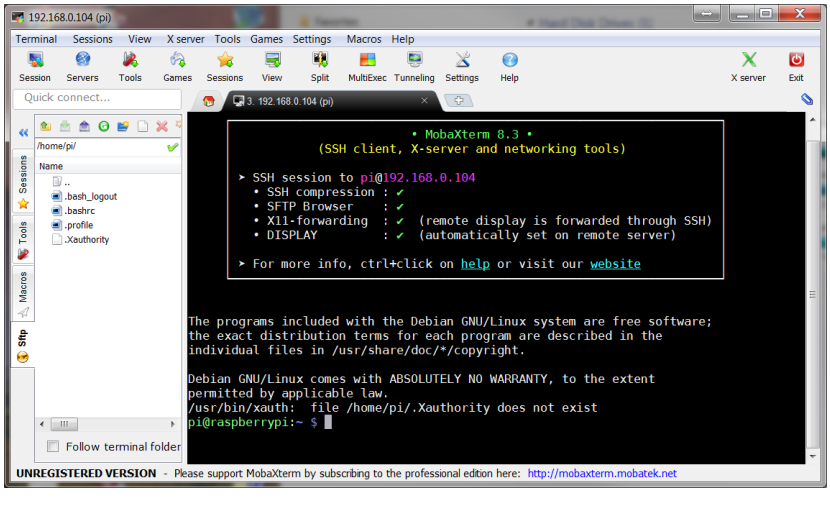

Figure 9: MobaXterm SSH Client

4.3.2 Resize Partition / FS

- On larger SD cards, the root partition can be resized to use extra space, using the *Expand Filesystem* option from raspi-config menu:

  $ sudo raspi-config

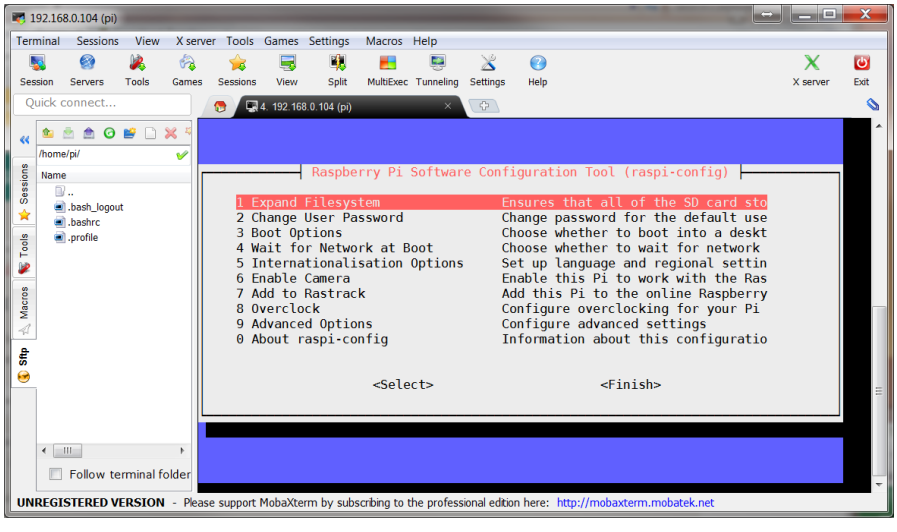

Figure 10: raspi-config Menu

- Select *1 Expand Filesystem* from raspi-config menu and press Enter:

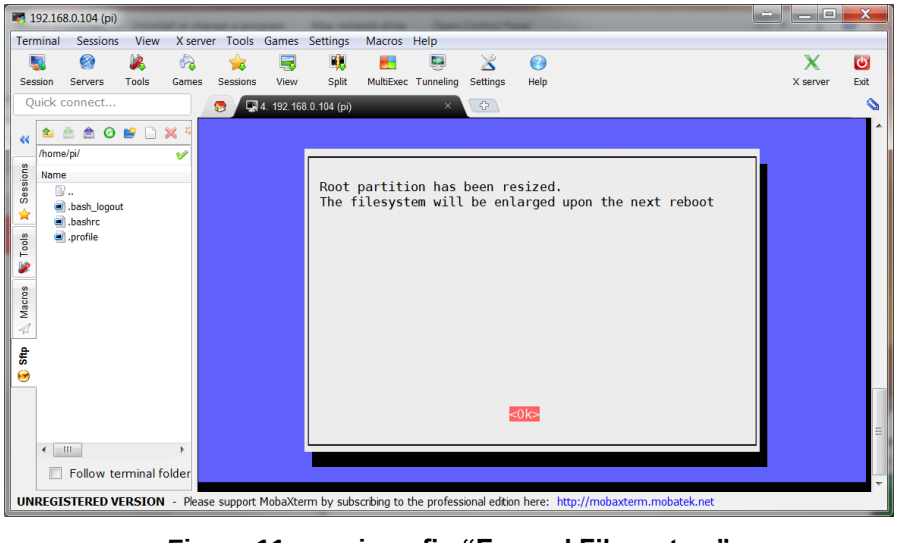

Figure 11: raspi-config "Expand Filesystem"

- The system must be then rebooted:

  $ sudo reboot

For more details, go to the following address:
https://www.raspberrypi.org/documentation/configuration/raspi-config.md

4.3.3 Update and configure the RPI

Update

Enter the following commands:

- *$* sudo apt-get update
- *$* sudo apt-get upgrade
- *$* sudo apt-get dist-upgrade
- *$* sudo rpi-update

Install Git

- *$* sudo apt install git

4.3.4 Clone Semtech HAL + Packet Forwarder

Get the latest Semtech software package from LoRa® Github (requires a connection to internet):

$ git clone https://github.com/Lora-net/gateway_2g4_hal

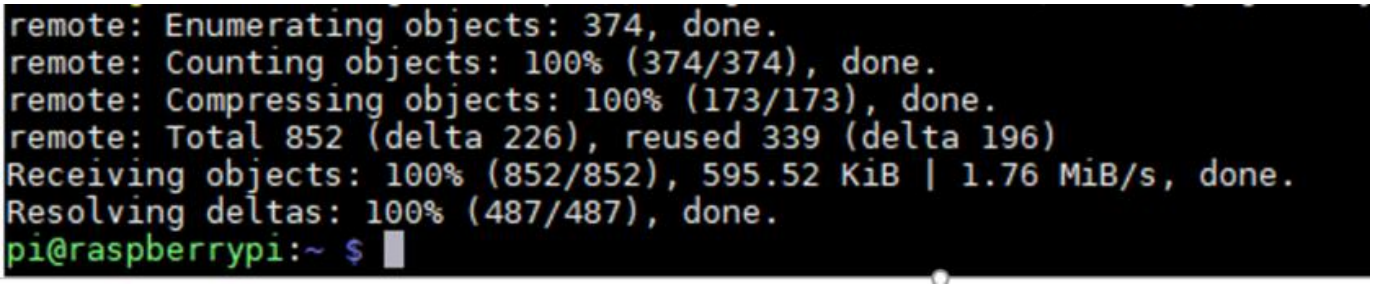

Figure 12: Git clone

4.3.5 Install dfu-util Tool

- $ cd ~/gateway_2g4_hal/
- $ sudo apt-get install autoconf
- $ git clone https://git.code.sf.net/p/dfu-util/dfu-util
- $ cd dfu-util
- $ ./autogen.sh
- $ sudo apt-get install libusb-1.0-0-dev
- $ ./configure
- $ make
- $ sudo make install

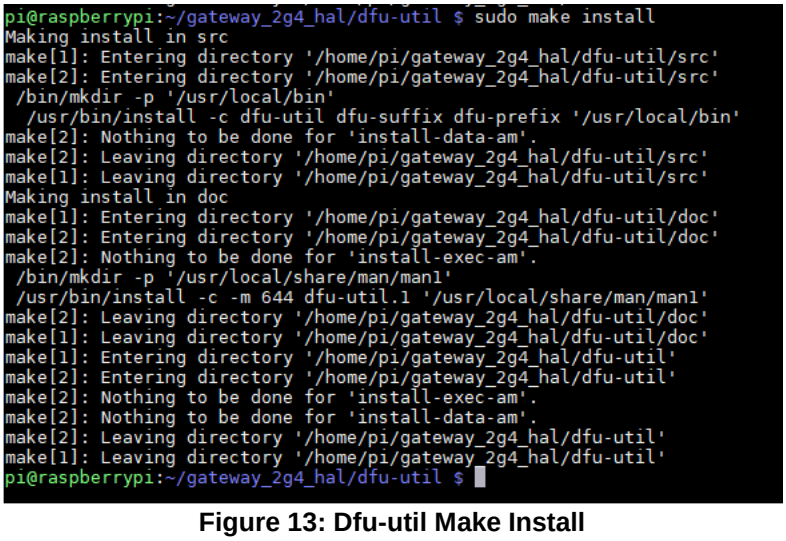

Figure 13: Dfu-util Make Install

4.3.6 Load STM32F446RC MCU Binary

For the first time only, load the STM32F446RC MCU binary:

1. Press the “BOOT0” button of the GW while resetting the gateway:

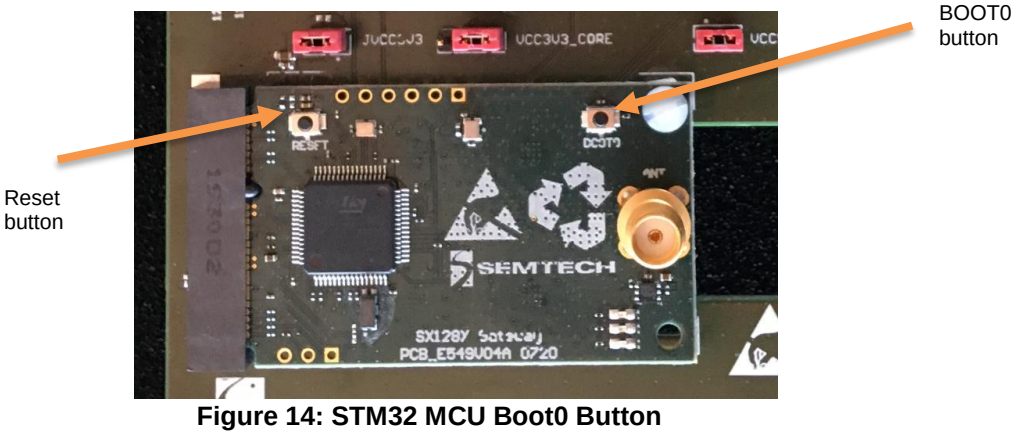

Figure 14: STM32 MCU Boot0 Button

2. On the Raspberry Pi, load the binary into the STM32F446RC MCU with the following command:

$ cd gateway_2g4_hal/dfu-util

$ sudo dfu-util -a 0 -s 0x08000000:leave -t 0 -D ../mcu_bin/rlz_fwm_gtw_2g4_00.02.16.bin

```
pi@raspberrypi:~ $ cd gateway_2g4_hal/dfu-util/
pi@raspberrypi:~/gateway_2g4_hal/dfu-util $ sudo dfu-util -a 0 -s 0x08000000:leave -t 0 -D ../mcu_bin/rlz_fwm_gtw_2g4_00.02.16.bin
dfu-util 0.9

Copyright 2005-2009 Weston Schmidt, Harald Welte and OpenMoko Inc.
Copyright 2010-2019 Tormod Volden and Stefan Schmidt
This program is Free Software and has ABSOLUTELY NO WARRANTY
Please report bugs to http://sourceforge.net/p/dfu-util/tickets/

dfu-util: Invalid DFU suffix signature
dfu-util: A valid DFU suffix will be required in a future dfu-util release!!!
Opening DFU capable USB device...
ID 0483:df11
Run-time device DFU version 011a
Claiming USB DFU Interface...
Setting Alternate Setting #0 ...
Determining device status: state = dfuIDLE, status = 0
dfuIDLE, continuing
DFU mode device DFU version 011a
Device returned transfer size 2048
DfuSe interface name: "Internal Flash  "
Downloading to address = 0x08000000, size = 74916
Download        [=========================] 100%        74916 bytes
Download done.
File downloaded successfully
Transitioning to dfuMANIFEST state
pi@raspberrypi:~/gateway_2g4_hal/dfu-util $
```

Figure 15: Load STM32F446RC MCU Binary

Note : potentially the MCU will be not up to date, it must be programmed with the latest version available into the mcu_bin folder.

For any future STM32F446RC MCU binary update, there should be no need to press the “BOOT0” button, simply connect the GW to any USB port of the Raspberry Pi and type following commands:

- *$ lsusb*

To check GW is recognized:

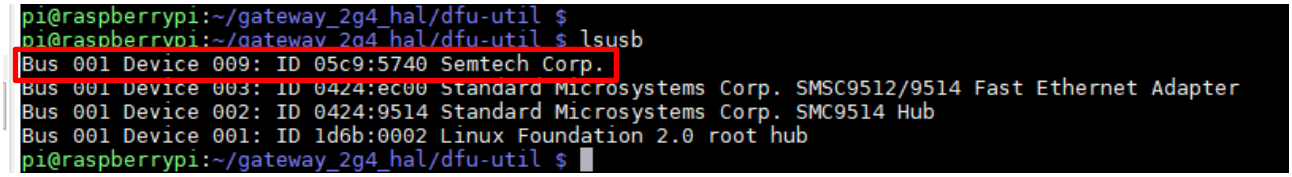

Figure 16: “lsusb” Command Result

- *$ dmesg*

To know the COM device to be used to access to the GW (here “ttyACM0”):

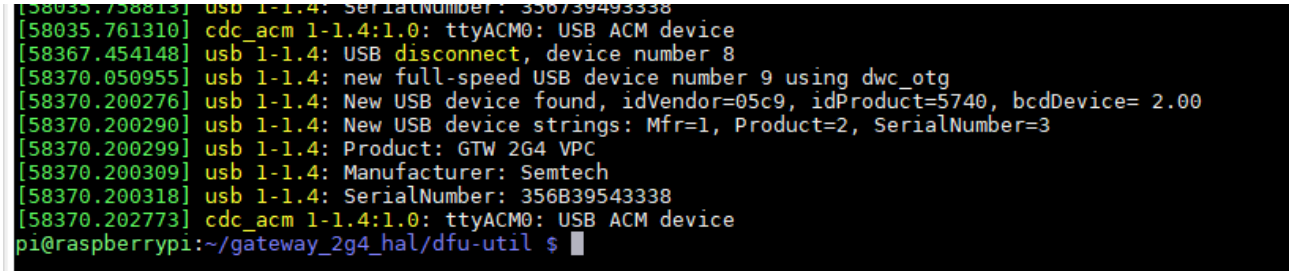

Figure 1712: “dmesg” Command Result

- $ cd ~/gateway_2g4_hal/util_boot
- $ make
- $ ./boot -d /dev/ttyACM0
- $ lsusb

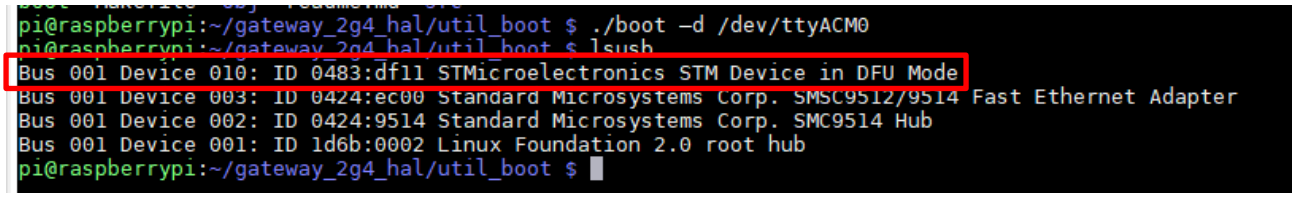

Figure 1813: DFU mode

$ cd ~/gateway_2g4_hal/dfu-util
$ sudo dfu-util -a 0 -s 0x08000000:leave -t 0 -D ../mcu_bin/rlz_fwm_gtw_2g4_01.00.01.bin

⇨ Your GW is updated !

4.3.7 Compile Semtech HAL + Packet Forwarder

- *$ cd ~/gateway_2g4_hal/*
- *$ make clean all*

The executables are copied in the different folders.

Test functions are in the libloragw folder:

```
pi@raspberrypi:~/gateway_2g4_hal/libloragw $ ls -l
total 528
drwxr-xr-x 2 pi pi  4096 Jan  9 09:33 inc
-rw-r--r-- 1 pi pi 72142 Jan  9 09:33 libloragw.a
-rw-r--r-- 1 pi pi   289 Jan  9 09:05 library.cfg
-rw-r--r-- 1 pi pi  1997 Jan  9 09:05 Makefile
drwxr-xr-x 2 pi pi  4096 Jan  9 09:33 obj
-rw-r--r-- 1 pi pi  8590 Jan  9 09:05 readme.md
drwxr-xr-x 2 pi pi  4096 Jan  9 09:05 src
-rwxr-xr-x 1 pi pi 69964 Jan  9 09:33 test_hal_cnt
-rwxr-xr-x 1 pi pi 68148 Jan  9 09:33 test_hal_reg
-rwxr-xr-x 1 pi pi 63448 Jan  9 09:33 test_hal_reset
-rwxr-xr-x 1 pi pi 71596 Jan  9 09:33 test_hal_rx
-rwxr-xr-x 1 pi pi 68388 Jan  9 09:33 test_hal_toa
-rwxr-xr-x 1 pi pi 76436 Jan  9 09:33 test_hal_tx
drwxr-xr-x 2 pi pi  4096 Jan  9 09:05 tst
pi@raspberrypi:~/gateway_2g4_hal/libloragw $
```

Figure 1914: libloragw executables

For instance:

Test_hal_tx : simple TX LoRa
Test_hal_rx : simple RX LoRa

Packet forwarder is in packet_forwarder folder

```
pi@raspberrypi:~/gateway_2g4_hal/packet_forwarder $ ls -l
total 196
-rw-r--r-- 1 pi pi   1344 Jan  9 09:05 global_conf.json
drwxr-xr-x 2 pi pi   4096 Jan  9 09:05 inc
-rwxr-xr-x 1 pi pi 149568 Jan  9 09:33 lora_pkt_fwd
-rw-r--r-- 1 pi pi   1551 Jan  9 09:05 Makefile
drwxr-xr-x 2 pi pi   4096 Jan  9 09:33 obj
-rw-r--r-- 1 pi pi  15651 Jan  9 09:05 PROTOCOL.md
-rw-r--r-- 1 pi pi  10737 Jan  9 09:05 readme.md
drwxr-xr-x 2 pi pi   4096 Jan  9 09:05 src
pi@raspberrypi:~/gateway_2g4_hal/packet_forwarder $
```

Figure 20: packet forwarder executable

Lora_pkt_fwd is the executable.

4.3.8 Semtech HAL Compilation Check

The program *test_hal_reg* is used to check the reliability of the link between the host platform and the LoRa® concentrator register file that is the interface through which all interactions with the LoRa® concentrator happen.

- *$ cd ~/gateway_2g4_hal/libloragw*
- *$ ./test_hal_reg*

The output looks like this:

```
pi@raspberrypi:~/gateway_2g4_hal/libloragw $ ./test_hal_reg
### LoRa 2.4GHz Gateway - Radio Register Read/Write ###
Read register 0x08C1:  0x21
Write register 0x08C1: 0xAA
Read register 0x08C1:  0xAA
### Exiting ###
pi@raspberrypi:~/gateway_2g4_hal/libloragw $
```

Figure 151: test_loragw_reg

4.3.9 Get the Unique ID to the Gateway

The 2.4GHz 3 Channels Single SF Reference Design has a unique ID given at production. This ID can be used as a 64-bit MAC address for the 2.4GHz 3 Channels Single SF Reference Design.

$ cd ~/gateway_2g4_hal/util_chip_id

$ ./chip_id

Return a unique ID like the following:

```
pi@raspberrypi:~/gateway_2g4_hal/util_chip_id $ ./chip_id
INFO: Channel 0 is disabled
INFO: Channel 1 is disabled
INFO: Channel 2 is disabled
## opening /dev/ttyACM0
INFO: Concentrator MCU version is V00.02.16

INFO: concentrator EUI: 0x16513833323930 35

## closing /dev/ttyACM0
pi@raspberrypi:~/gateway_2g4_hal/util_chip_id $
```

Figure 16: util chip ID

The gateway ID could be then replaced in the global_conf.json file within the repository:

~/gateway_2g4_hal/packet_forwarder/global_conf.json

⇨ *The chip id unicity is no guaranteed yet, please choose your own waiting for an improvement*

```
"gateway_conf": {
    "gateway_ID": "1651383332393035", // Do not let it in auto
    /* change with default server address/ports */
    "server_address": "semtech.eu1.cloud.thethings.industries",
    "serv_port_up": 1700,
    "serv_port_down": 1700,
    /* adjust the following parameters for your network */
    "keepalive_interval": 10,
    "stat_interval": 30,
    "push_timeout_ms": 100,
…
…
…
```

4.3.10 Test the HAL TX

The program *./test_loragw_hal_tx* is used to test the emission of the 2.4GHz GW 3+1 reference design with settings set by the user.

The tests run endlessly or until an error is detected: press Ctrl+C to stop the application.

- *$ cd ~/gateway_2g4_hal/libloragw*
- *$ ./test_hal_tx -f 2402 -s10 -b800 -l24 -n10000 -z24 -p13 -t 100*

The command above send a LoRa frame at 2.402 GHz and the power from the SX1280 set to 13dBm (-p).

For more information, enter:

$ ./test_loragw_hal_tx -h

4.3.11 Run Packet Forwarder

The Packet Forwarder is a program running on the host of a LoRa® Gateway that forward RF packets received by the concentrator to a server through an IP/UDP link, and emits RF packets that are sent by the server.

Run Packet Forwarder for a functional check:

$ cd ~/gateway_2g4_hal/packet_forwarder

$ ./lora_pkt_fwd

The output looks like this:

```
pi@raspberrypi:~ $ cd gateway_2g4_hal/packet_forwarder/
pi@raspberrypi:~/gateway_2g4_hal/packet_forwarder $ ./lora_pkt_fwd
*** Packet Forwarder 2.4Ghz***
Version: 0.2.0
*** HAL library version info ***
Version: 0.2.0;
***
INFO: Little endian host
INFO: found configuration file global_conf.json, parsing it
INFO: global_conf.json does contain a JSON object named radio_conf, parsing radio parameters
INFO: tty_path /dev/ttyACM0
INFO: antenna_gain 0 dBi
INFO: channel 0 enabled: frequency 2403000000, bandwidth 812000Hz, SF12, RSSI offset 0.0
INFO: Setting channel 0 configuration => en:1 freq:2403000000 sf:12 bw:800khz rssi_offset:0.0
INFO: channel 1 enabled: frequency 2479000000, bandwidth 812000Hz, SF12, RSSI offset 0.0
INFO: Setting channel 1 configuration => en:1 freq:2479000000 sf:12 bw:800khz rssi_offset:0.0
INFO: channel 2 enabled: frequency 2425000000, bandwidth 812000Hz, SF12, RSSI offset 0.0
INFO: Setting channel 2 configuration => en:1 freq:2425000000 sf:12 bw:800khz rssi_offset:0.0
INFO: TX enabled, freq min 2400000000Hz, freq max 2500000000Hz
INFO: Setting TX Enabled
INFO: global_conf.json does contain a JSON object named gateway_conf, parsing gateway parameters
INFO: gateway MAC address is configured to 1651383332220290
INFO: server hostname or IP address is configured to "ism2400.demo.thethings.industries"
INFO: upstream port is configured to "1700"
INFO: downstream port is configured to "1700"
INFO: downstream keep-alive interval is configured to 10 seconds
INFO: statistics display interval is configured to 30 seconds
INFO: upstream PUSH_DATA time-out is configured to 100 ms
INFO: packets received with a valid CRC will be forwarded
INFO: packets received with a CRC error will NOT be forwarded
INFO: packets received with no CRC will NOT be forwarded
## opening /dev/ttyACM0
INFO: Concentrator MCU version is V00.02.16
INFO: Configuring RX channel 1 => freq:2479000000 sf:12 bw:800khz
INFO: Configuring RX channel 2 => freq:2425000000 sf:12 bw:800khz
INFO: Configuring RX channel 0 => freq:2403000000 sf:12 bw:800khz
INFO: [main] concentrator started, packet can now be received
INFO: concentrator EUI: 0x1651383332393035
INFO: [down] PULL_ACK received in 34 ms
INFO: [down] PULL_ACK received in 33 ms
INFO: [0] Number of packets received with CRC OK:  1
src/lora_pkt_fwd.c:1119:thread_up():
Current time: 2020-01-30 10:08:06 GMT

INFO: Received pkt from mote: 0018A462 (fcnt=597)

CH0:    0       0       0       0       0       0       0       1
CH1:    0       0       0       0       0       0       0       0
CH2:    0       0       0       0       0       0       0       0

Total number of LoRa packet received: 1
Total number of LoRa packet sent: 0

JSON up: {"rxpk":[{"jver":1,"tmst":137822503,"chan":0,"freq":2403.000000,"foff":91,"stat":1,"modu":"LORA","datr":"SF12BW812","co
dr":"4/7LI","lsnr":-18.0,"rssi":-105,"size":23,"data":"QGKkGAAAVQIBgS404O8VWpPDonObJiA="}]}
INFO: [up] PUSH_ACK received in 35 ms
```

Figure 23: Packet Forwarder

5 JSON file for RF Parameter Tuning

Edit the file *~/gateway_2g4_hal/packet_forwarder/global_conf.json* update the following RF parameters:

- *freq*, to set frequency channels
    - Frequency channels in Hz
- *Bandwidth*, to set bandwidth channels
    - bandwidth channels in Hz
- *Spreading factor*, to set spreading factor channels
    - Spreading factor between 7 and 12

A typical LoRa® 2.4 GHz GW 3+1 reference design *global_conf.json* file looks like this:

```
{
   "radio_conf": {
      "tty_path": "/dev/ttyACM0",
      "antenna_gain": 0, /* antenna gain, in dBi */
      "chan_0": {
         "enable": true,
         "freq": 2403000000,
         "bandwidth": 812000,
         "spread_factor": 12,
         "rssi_offset": 0.0
      },
      "chan_1": {
         "enable": true,
         "freq": 2479000000,
         "bandwidth": 812000,
         "spread_factor": 12,
         "rssi_offset": 0.0
      },
      "chan_2": {
         "enable": true,
         "freq": 2425000000,
         "bandwidth": 812000,
         "spread_factor": 12,
         "rssi_offset": 0.0
      },
      "tx": {
         "enable": true,
         "tx_freq_min": 2400000000,
         "tx_freq_max": 2500000000
      }
   },

   "gateway_conf": {
      "gateway_ID": "1651383332393022",
      /* change with default server address/ports */
      "server_address": "semtech.eu1.cloud.thethings.industries",
      "serv_port_up": 1700,
      "serv_port_down": 1700,
      /* adjust the following parameters for your network */
      "keepalive_interval": 10,
      "stat_interval": 30,
      "push_timeout_ms": 100,
      /* forward only valid packets */
      "forward_crc_valid": true,
      "forward_crc_error": false,
      "forward_crc_disabled": false
   }
}
```

6 References

[1] 2.4GHz 3 Channels Single SF Reference Design information:
https://www.semtech.com/products/wireless-rf/lora-gateways/SX1280ZxxxxGW1

7 Part Number

[1] 2.4GHz 3 Channels Single SF Reference Design development kit Part Number : **SX1280Z3DSFGW1**

8 Revision History

Version	Date	Modifications
1.0	April 2020	First Release

9 Glossary

BB	BaseBand
BoM	Bill Of Materials
BW	BandWidth
CLK	Clock
CW	Continuous Wave
ETSI	European Telecommunications Standard Institute
DFU	Device Firmware Update
EU	Europe
EUI	Extended Unique Identifier
GB	GigaByte
GPS	Global Positioning System
GW	GateWay
HAL	Hardware Abstraction Layer
HDMI	High-Definition Multimedia Interface
HW	HardWare
IP	Intellectual Property
ISM	Industrial, Scientific and Medical applications
LAN	Local Area Network
LBT	Listen Before Talk
LO	Local Oscillator
LoRa®	LOng RAnge modulation technique
LoRaWAN	LoRa® low power Wide Area Network protocol
LPF	Low Pass Filter
LSB	Least Significant Bit
LUT	Look Up Table
MAC	Media Access Control address
MCU	Micro-Controller Unit
MPU	Micro-Processing Unit
PA	Power Amplifier
RSSI	Received Signal Strength Indication
RF	Radio-Frequency
RX	Receiver
SAW	Surface Acoustic Wave filter
SD Card	Secure Digital Card
SF	Spreading Factor
SPI	Serial Peripheral Interface
SPDT	Single-Pole, Double-Throw switch
SSH	Secure SHell
SW	SoftWare
TX	Transmitter
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
USB	Universal Serial Bus

Important Notice

Information relating to this product and the application or design described herein is believed to be reliable, however such information is provided as a guide only and Semtech assumes no liability for any errors in this document, or for the application or design described herein. Semtech reserves the right to make changes to the product or this document at any time without notice. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Semtech warrants performance of its products to the specifications applicable at the time of sale, and all sales are made in accordance with Semtech's standard terms and conditions of sale.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS, OR IN NUCLEAR APPLICATIONS IN WHICH THE FAILURE COULD BE REASONABLY EXPECTED TO RESULT IN PERSONAL INJURY, LOSS OF LIFE OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

The Semtech name and logo are registered trademarks of the Semtech Corporation. All other trademarks and trade names mentioned may be marks and names of Semtech or their respective companies. Semtech reserves the right to make changes to, or discontinue any products described in this document without further notice. Semtech makes no warranty, representation or guarantee, express or implied, regarding the suitability of its products for any particular purpose. All rights reserved.

© Semtech 2020

Contact Information

Semtech Corporation
Wireless & Sensing Products
200 Flynn Road, Camarillo, CA 93012
E-mail: sales@semtech.com
Phone: (805) 498-2111, Fax: (805) 498-3804
www.semtech.com