IoTLabs: Exploring LoRa Technology

As defined by Semtech, LoRa is *a wireless technology developed to create the low-power, wide-area networks (LPWANs) required for machine-to-machine (M2M) and Internet of Things (IoT) applications. The technology offers a very compelling mix of long range, low power consumption and secure data transmission and is gaining significant traction in IoT networks being deployed by wireless network operators*.

In this lab, you will implement a prototype of LoRa communication between two wireless devices. This enables you to get hands-on experience with LoRa, assess the radio performance, and prepare future advanced prototypes and experimentations.

- What are the advantages of the LoRa modulation?
- How LoRa is compatible with LPWAN requirements and constraints?

-. Setting the Lab

-. Hardware Platform

In order to design and implement experiments with LoRa, you will use the following devices:

- Arduino Mega (x2).
- LoRa shields from Dragino (x2).

- Give the characteristics of the Arduino you are using: model, number of pins, type of pins, memory sizes, etc.
- Give the main characteristics of the LoRa shield from Dragino (www.dragino.com).
- What type of Antenna are you using? Explain the corresponding characteristics.
- Give an estimated cost of your platform.

-. Software Tools

Download the following software on your PC:

- RadioHead: The Packet Radio library for embedded microprocessors can be downloaded from https://github.com/samerlahoud/RadioHead.
- Arduino IDE: Specific OS versions can be downloaded from https://www.arduino.cc/en/Main/Software.

Unzip the RadioHead library and copy it to your sketchbook library folder as detailed in https://www.arduino.cc/en/Guide/Libraries.

Note well the location of the library folder on your computer. In the following, you will be required to modify source files located in this folder.

-. Installation

Start by plugging the Dragino shields on the Arduino devices and mounting the antennas as shown in Fig. 1.

Figure 1. Arduino with LoRa Dragino shield.

Connect the two Arduino devices to USB ports on your computer. If this is the first time you use Arduino IDE, make sure to install the necessary USB drivers by selecting `Tools > Boards Manager` and installing Arduino AVR boards.

Now, you have to choose the `Board` type as `Arduino/Genuino Mega 2560` in the `Tools` menu and select the corresponding serial `Port` to start programming your Arduino.

For Arduino Mega 2560, additional drivers for Microsoft Windows can be installed from http://wch.cn/download/CH341SER_ZIP.html.

-. Theoretical Study

In this section, you will perform a theoretical assessment of the performance of LoRa modulation. You will later compare this theoretical results to the experimental ones as in a typical scientific study.

- What is the relation between processing gain and spreading factor in LoRa modulation? Explain.
- How does the spreading factor impact the coverage of a LoRa transmitter?
- For each of the three possible configurations of your LoRa device, what is the

transmission bit rate? Explain your computation.
- Compute the receiver sensitivity, assuming the following parameters: channel bandwidth = 125 kHz, spreading factor = 7, coding rate = 4/5, bit error rate (BER) target = 10^{-4}, and receiver noise figure = 6 dB. Refer to this article to determine the mapping between the BER and the SNR.
- Compare the computed sensitivity to that provided by the Semtech Application Note AN1200.22 for the same parameters.

In the remainder of this lab, you will conduct measurements to validate the obtained theoretical receiver sensitivity.

-. Configuring and Running the Lab

-. Modifying the Radio Parameters

Download the basic sketches that implement a simple LoRa communication between the two devices: a client and a server. Open the sketches with Arduino IDE. Make sure to choose the correct `Board` and `Port` in the `Tools` menu.

Take a look at the source code in `rf95_client.ino` and `rf95_server.ino`. Particularly, the `setup` function configures the radio parameters of your LoRa devices:

- Central frequency (freq)
- Spreading Factor (SF)
- Bandwidth (Bw)
- Coding Rate (CR)
- Transmit power (Pow)

```
rf95.setFrequency(frequency);
// Setup Power,dBm
rf95.setTxPower(13);

// Setup Spreading Factor (6 ~ 12)
rf95.setSpreadingFactor(7);

// Setup BandWidth, option:
7800,10400,15600,20800,31250,41700,62500,125000,250000,500000
//Lower BandWidth for longer distance.
rf95.setSignalBandwidth(125000);

// Setup Coding Rate:5(4/5),6(4/6),7(4/7),8(4/8)
rf95.setCodingRate4(5);
```

In order to reduce collisions, configure the central frequency of your LoRa devices as indicated below:

Group Number	Frequency
1	866.7
2	866.9
3	867.1
4	867.3
5	867.5
6	867.7
7	867.9
8	868.1
9	868.3
10	868.5
11	868.7
12	868.9

-. Running Basic Sketches

Now you can compile and upload the client and server sketches on the two arduino devices, respectively. On the serial interfaces, you should obtain similar results as in Fig. 2 and Fig. 3. The client sends periodically a short message towards the server. The server outputs the RSSI (received power in dBm) for each received message.

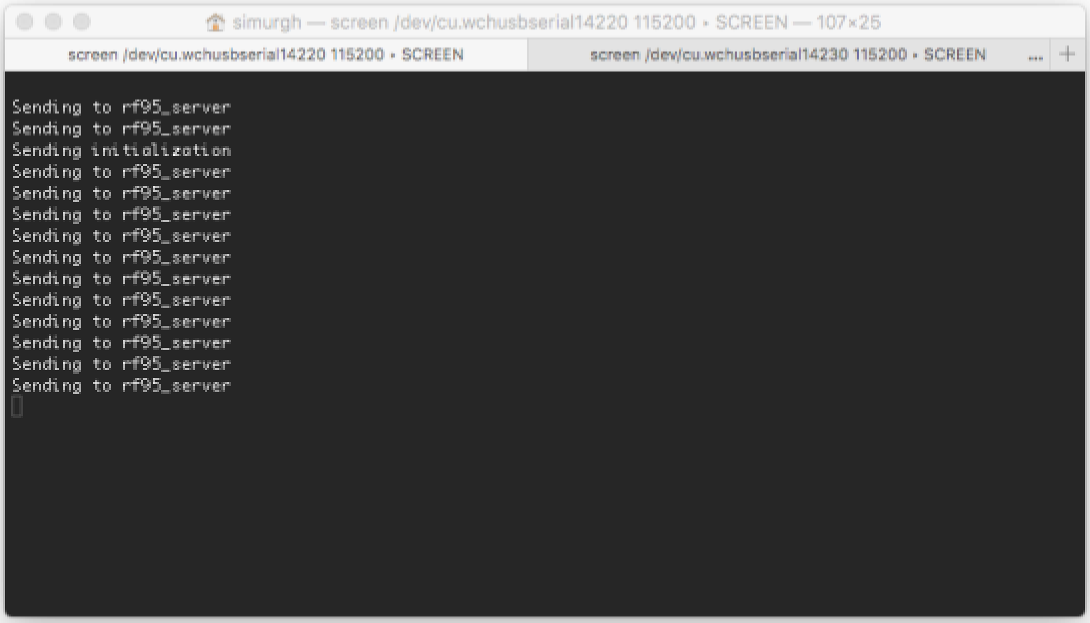

Figure 2. Client serial monitor

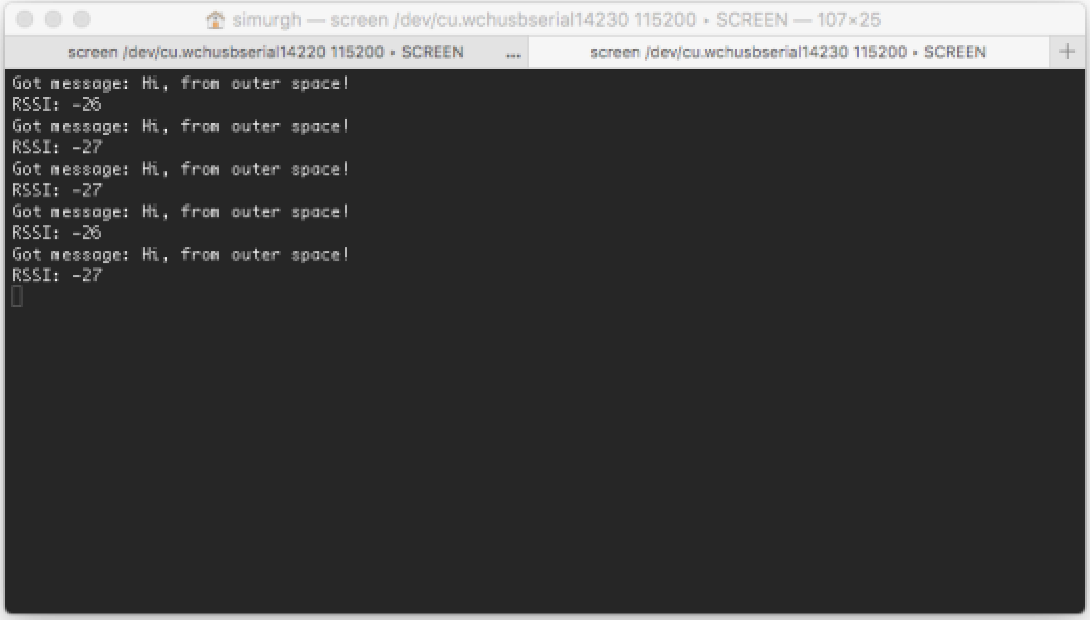

Figure 3. Server serial monitor

-. Performance Evaluation

In the following, you will design and implement a set of scenarios that enable to evaluate the performance of the LoRa modulation. As you will deal with scientific assessment, you are required to use scientific tools to show the results. You have the choice between gnuplot, matplotlib with Python, and MATLAB. Take some time to become familiar with one of these software as you will be required to use them in different occasions of your academic programme.

As we are in presence of variable radio conditions, some experiments should be repeated multiple times and results can be shown as probability distributions. Take a look at this excellent repository of data visualisation tools https://www.data-to-viz.com.

-. Time on Air

In this section, you will measure the Time on Air (ToA) as given by the time necessary to transmit a message on the radio interface. You will assess the impact of the spreading factor, bandwidth, coding rate, and the message size on the ToA.

For this, you will start by implementing a function on the client that measures the time necessary for sending a message. For example, you can use the micros() function available in the arduino libraries. Now, you can modify one of the parameters (spreading factor, bandwidth, coding rate, message size) and record the impact on the ToA. Note well that you may need to repeat the expriment to obtain the statistical distributions.

- Describe the scenarios you used for assessing the impact of the different parameters on the ToA.

- Join commented extracts of your code and raw data in attached files.
- Visualise the experimental results by plotting the ToA as a function of each one of the different parameters.
- Analyze the obtained results and compare with the theoretical computations. You can superpose the theoretical results and the experimental ones on the same graph.

-. [Classroom activity] Collisions and Packet Delivery Ratio

In this section, you will measure the Packet Delivery Ratio (PDR) under different transmission periods. Only for this test, all groups are required to use the same frequency, spreading factor, and coding rate.

You will start by implementing a function on the server that measures the ratio of successfully delivered packets.

- Draw the PER as a function of the transmission period for the different radio configurations. Analyze your results.
- What type of mathematical models enables to theoretically compute the PER?

-. Coverage

In this section, you will measure the coverage of LoRa under the three different radio configurations. Such configurations should ensure different reliability levels.

For this, you will start by identifying a set of Test Points (TP). Then, you should implement a function that sends packets with different radio configurations. Note that the following functions in the Arduino sketch enable to modify *on the fly* the LoRa parameters:

```
rf95.setModemConfig(RH_RF95::Bw125Cr45Sf128);
rf95.setModemConfig(RH_RF95::Bw31_25Cr48Sf512);
rf95.setModemConfig(RH_RF95::Bw125Cr48Sf4096);
```

- Draw the test points on a map.
- Give a statistical measure of the PER and the RSSI for each TP with each of the different radio configurations.

-. Path Loss

In this section, you will study the properties of the radio channel as used by the LoRa technology. For this, you should obtain a large set of RSSI values for different distances, preferably in a free space setting.

- Provide the expression of the link budget.
- Draw the RSSI values as a function of the distance.
- What is the path loss exponent?
- Using regressions, compute a value of the path loss exponent. Analyze the result.
- Compare the obtained sensitivity with the theoretical results computed in the previous section.

In order to compute distances in your experiment, you can get the GPS coordinates as recorded by your smartphone using an application such as Ultra GPS Logger. You can export the time-location correspondence in a CSV format from this application. As for the time-RSSI correspondence, you can use a

logger file

on your laptop. Finally, the time matching enables you to obtain the RSSI for each GPS location, hence for different distances.

-. Coverage Challenge

-. Grading

	Exemplary	**Accomplished**	**Developing**	**Beginning**
Answer to questions				
Design experiments				
Analyse results				