CSC2006
Internet of Things: Protocols and Networks

LABORATORY 1 MANUAL
LoRa

AY 2022/23

OBJECTIVES

- To familiarise with LoRa.
- Develop a LoRa-based peer-to-peer solution on an Arduino and Cytron LoRa-RFM Shield.

EQUIPMENT

- Computer / Laptop
- Arduino Uno
- Cytron LoRa-RFM Shield

INTRODUCTION

LoRa (short for long-range), is an RF modulation technique that allows low-powered wireless communication systems. Since the frequencies used are orders of magnitude smaller than other standard wireless communication methods (i.e. Wi-Fi), the transmission range is much larger (i.e. up to 16 kilometres in rural areas with a clear line of sight). Coupling this long-range transmission with low power requirements makes LoRa a good choice for battery-powered Internet of Things (IoT) devices with applications requiring communication with widely-distributed systems. LoRa networks use a star topology, with a central gateway connecting multiple end devices. This makes it easy to scale the network as needed, enabling the deployment of large-scale IoT networks. Finally, LoRa operates in the license-free ISM frequency band, making it a cost-effective solution for IoT applications. Additionally, LoRa devices are relatively inexpensive, making it possible to deploy large numbers of devices cost-effectively.

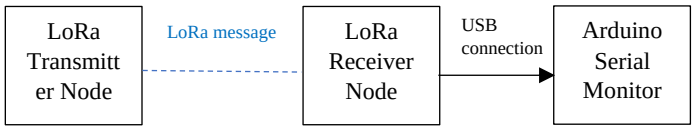
In summary, LoRa provides long-range connectivity, low power consumption, scalability, security, and cost-effectiveness, making it an essential technology in the growth and development of the IoT.

The goal of this lab session is for students to gain hands-on experience setting up and using LoRa devices in a peer-to-peer (P2P) configuration and to understand the potential for using LoRa in IoT applications.

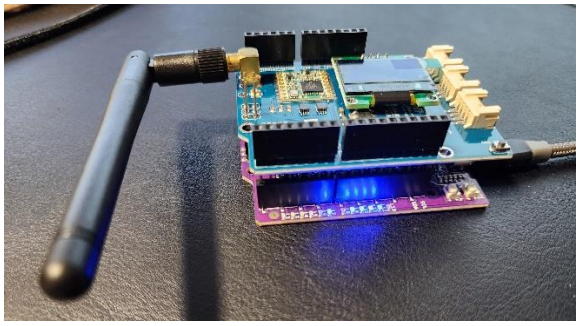
EXERCISE 1: LoRa

In this exercise, the **Maker UNO** coupled with the **Cytron LoRa-RFM Shield** will build a LoRa-based end node to exchange data between two nodes in a per-to-peer configuration. We will set up a transmitter node which will broadcast a LoRa payload containing some random data at regular intervals. Then, we will build a LoRa receiver node that will receive the payload and display it on the serial monitor.

In addition to the hardware that has been given to you, you will need to install the serial driver for the Maker UNO found here [3]. The LoRa library should also be installed



Ensure that the Arduino UNO is connected to the LoRa Shield, as shown below.



1. Setup the Arduino IDE
   a. If you do not have Arduino IDE, download and install the latest IDE.
      i. I would suggest version 1.8x
      ii. Download and install the CH341 driver for the Arduino UNO.
   b. We need to install the open-source library RadioHead by Paul Stoffregen. This will allow us to send and receive data using LoRa-based radios. The library (zip file) can be downloaded from the following site.
   c. At the Arduino IDE menu, select **Tools > Board > Arduino Uno**.
   d. Connect the Arduino to the computer's USB port. At the Arduino IDE menu, select **Tools > Port**. We need to select the Serial Port that the Arduino is connected to.

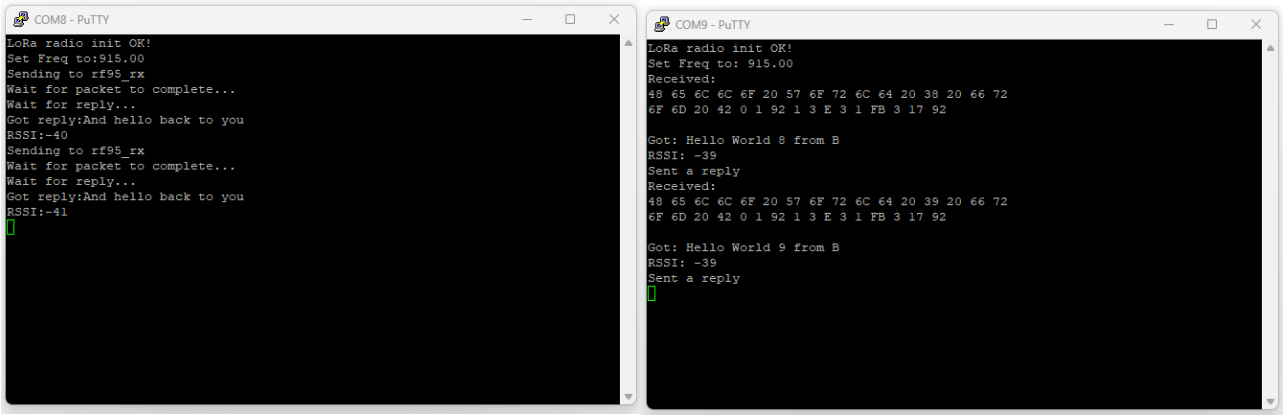
2. Use the examples given in the xSITe to program the LoRa-based P2P communication.
   a. Lab 3 - rf95_rx.ino (Receiver)
   b. Lab 3 - rf95_tx.ino (Transmitter)

3. Verify and upload the sketch.
   a. Verify the Sketch to check for any errors.
   b. Next, upload the Sketch to the device. When the upload has taken place, a message "Done uploading" is seen at the end of the sketch file window.

4. Start the **Transmitter** Node. Open the Serial Monitor to observe the output.

5. Open the Serial Monitor on the **Receiver** Node. You should see the data updated each time it receives a message.

a. Select **Tools > Serial Monitor**. The Serial Monitor window will open.
b. Select the **9600 baud** option from the drop-down menu at the bottom-right of the Serial Monitor.



Lab Assignment

1. Integrate and display the following information on the OLED the status of the LoRa Receiver & LoRa Sender. You may refer to "*Lab 3 - ssd1306_i2c.ino*" to learn how to display data on the OLED screen. Do remember to install the "*Adafruit SSD1306*" library before compiling the code.
    a. "Setup Successful"
    b. "Setup Failed"
    c. "Sending Message"
    d. "Waiting for Reply"
    e. "Message Received"

2. To improve the reliability of the peer-to-peer communication that will not be affected by crosstalk traffic, e.g. ACK and retransmit.

3. To improve ID-based messages and protocols with a header, payload and checksum that can support at least three devices/parties (i.e. simple message filtering, only accepting messages that are directed to the node's agreed ID)

REFERENCE

[1] SemTech, https://www.semtech.com/lora/what-is-lora
[2] SemTech, Building a LoRa®-based Device End-to-End with Arduino | DEVELOPER PORTAL (semtech.com)
[3] CH341 Serial Driver, https://www.wch.cn/downloads/CH341SER_ZIP.html