TALLINN UNIVERSITY OF TECHNOLOGY

Thomas Johann Seebeck Department of Electronics

LoRa Based Snow Depth Detection

Supervisor:

Dr.Alar Kuusik

Members of the group:

Harish Kumar Singh (177319IVEM)

TALLINN 2016

Table of Contents

Abstract

The aim of the project is to measure snow depth and send this information to any server over LoRa. The snow depth detection module “512A Laser Distance Meter” is available. I have achieved following goals in this project:

1. Configure Multitech Conduit LoRa Gatway to receive packet from LoPy in ABP mode and send these packet to TTN server.

2. Configure TTN server to receive LoRa packet from Conduit Gateway.

3. Interface snow depth detection module to LoPy over UART.

4. Write script in Python for LoPy to read sensor data over UART and send it to LoRa gateway.

1. Introduction

I have used LORA technology for transmission of data from Lora Gatway to server. Below are the different module used in this project:

Snow detection module: 512A Laser Distance Meter
Lora Node: LoPy
Lora Gateway/Base-Station: Multitech Conduit
Server/Cloud: The Thing Network

2. Development Environment

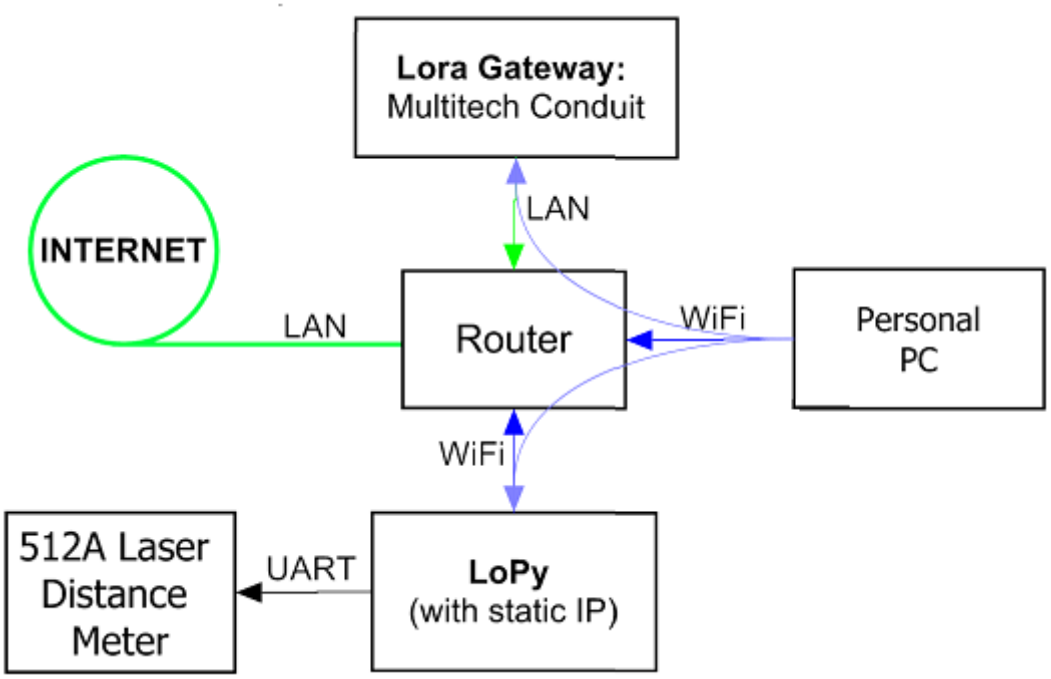

Figure 1: Developing Environment

LoPy is configured with static IP and Conduit is enabled with DHCP client to have accessible IP address on connecting them to router. I use ATOM IDE to programme LoPy and Putty to access Conduit.

3. Development Steps

Aim is to bring LoPy, Conduit and Personal PC in local network. So, it will be easy to access LoPy and Conduit Gatway.

Step 1: Configure LoPy with static IP. Write script to connect LoPy to WiFi router. Add script to LoPy boot up sequence. So LoPy connect to WiFi on

power on [1][2].

Step 2: Enable Conduit DHCP Client to add it within local network [3][4].

Step 3: Register Gateway on TTN network. This will give Gateway ID and Key.

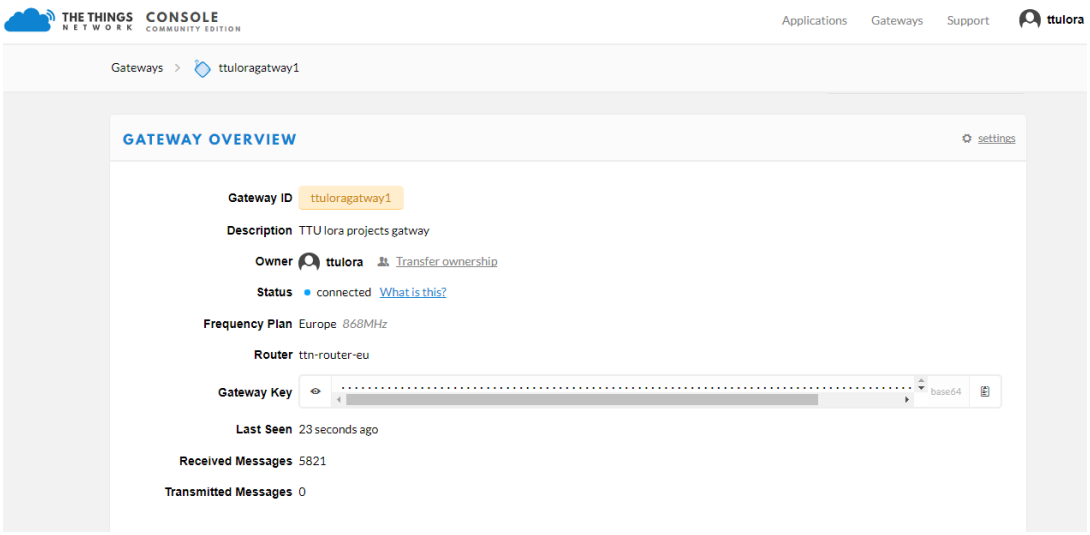

Figure 2: TTN Gateway configuration

Step 4: Install the TTN Packet Forwarder on a Multitech Conduit and configure it with Gateway ID and Key generated from pervious step [3][5][6].

Step 5: On TTN, create application which will give Application ID, Application EUIS and Access Keys. Then also create device in ABP (Authentic ion by personalization) mode. This will give Device ID, Device EUI, Application EUI, Device Address, Network Session Key, App Session Key.

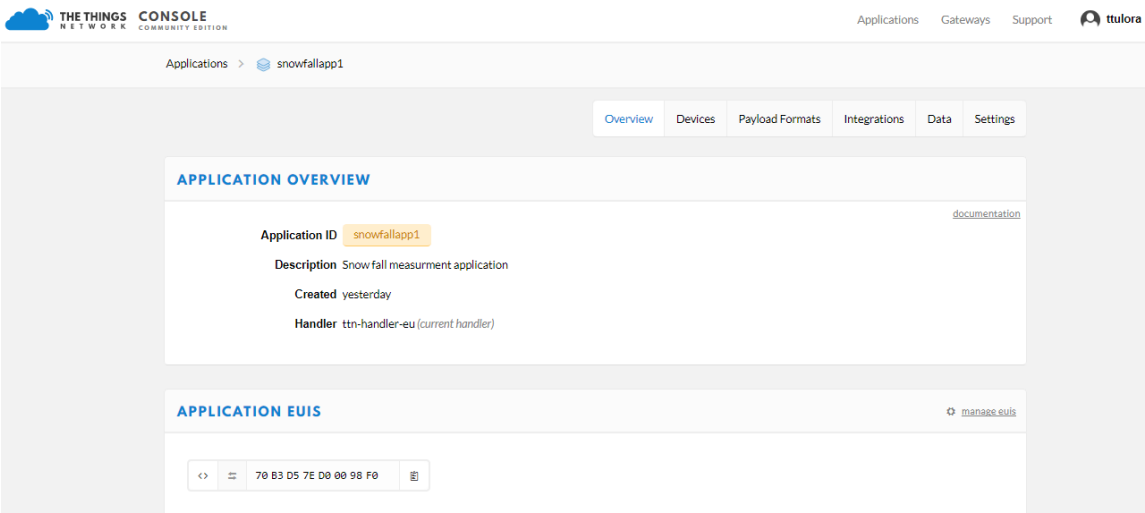

Figure 3: TTN Application configuration

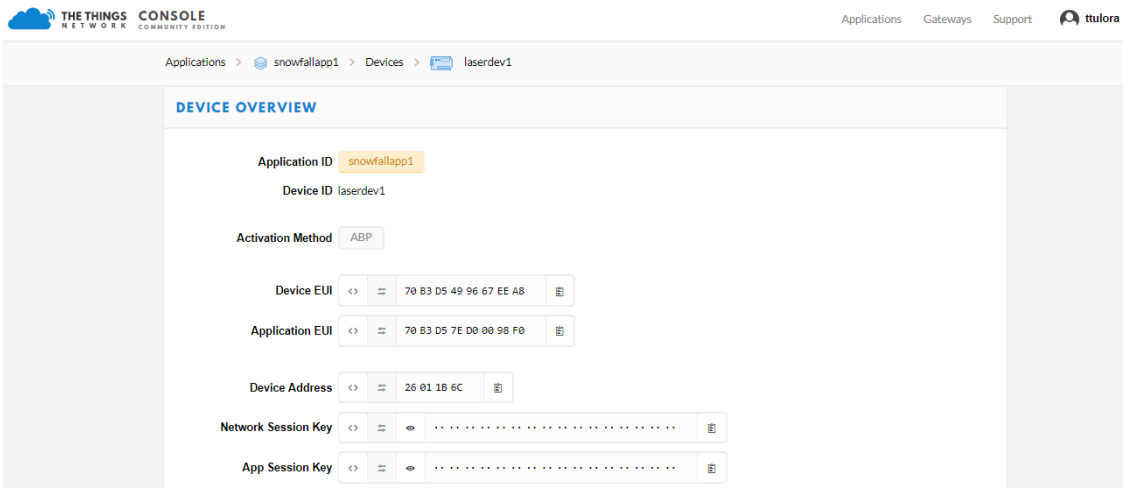

Figure 4: TTN Device configuration

Step 6: Write script on LoPy to initialize LoRa module in ABP mode and send data packet to Conduit Gate. Use Device Address, Network Session Key, App Session Key from pervious step.

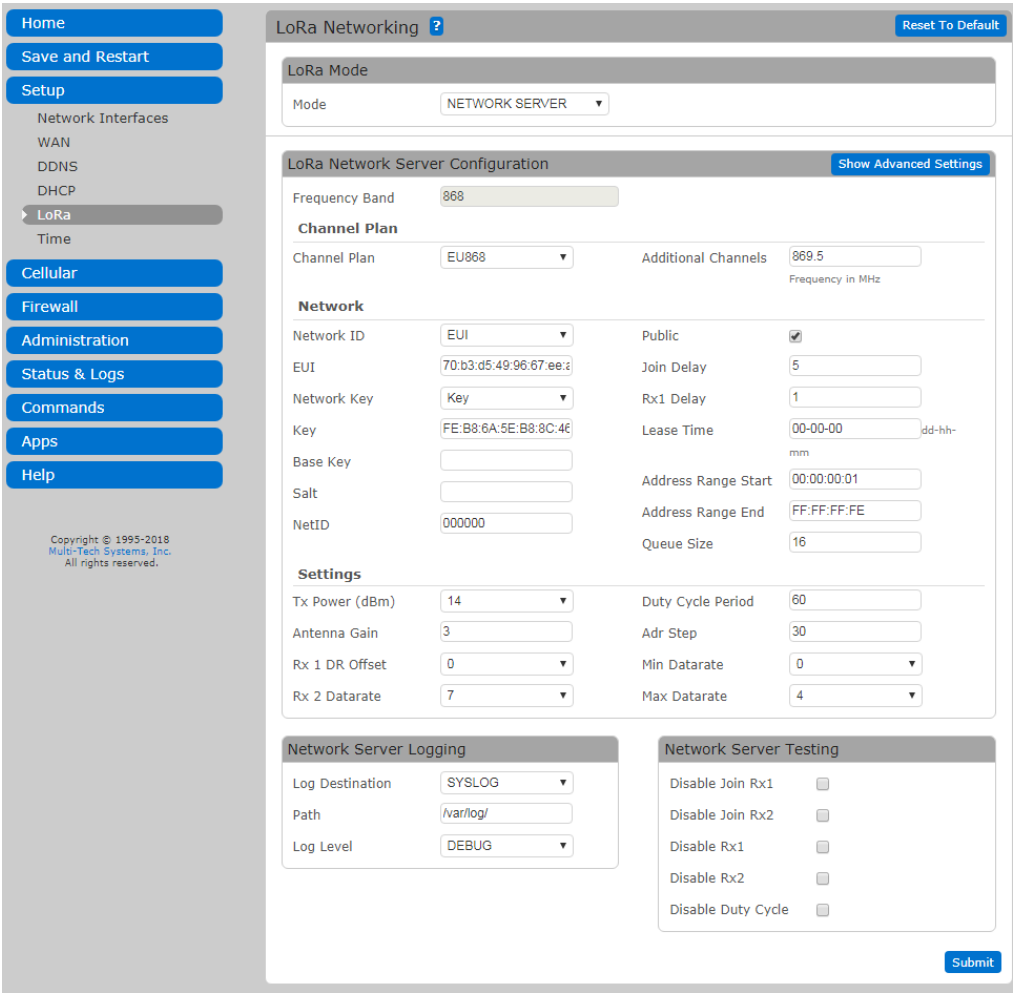

Figure 5: Conduit LoRa configuration

Step 7: Configure conduit Lora module. Use Device Address and App Session Key from step 5, as shown in Figure 5.

Step 8: Connect 512A Laser Distance Meter to LoPy over UART. LoPy expansion board P3 and P4 is TXD and RXD respectively. Write script to access distance data and encode in with following format:

```
datapack.insert( 0,(distval >> 8)  )
datapack.insert( 1,(distval & 0xFF))
```

This will encode 5 digit number with 3 decimal place in 2 bytes. Decode this packet in TTN device using following script:

```
function Decoder(bytes, port) {
  var distance = (bytes[0] << 8) | bytes[1];
  return {
    meters: distance / 1000.0
  };
}
```

Step 9: On LoPy, integrate UART and LoRa script. Add this script to LoPy boot up sequence.

4. Installation

Follow following steps for installation:

1) If Conduit is connected directly to LAN-Port then change Conduit configuration from LAN to WAN (DHCP Client). If Conduit is connected to router over LAN as show in Figure 1, then keep Conduit configured to LAN (DHCP Client).
2) Power on the LoPy device. Let Led start-up sequence finish
3) Connect 512A Laser Distance Meter module to LoPy

Laser module must be connected after Led start-up sequence finish.

5. Simulation

Data will be sent at ever 45 sec. Figure 6 show TTN gateway traffic with device address and other packet information. Figure 7 show distance measurement in meters. TTN can't store data packets. So we can see data only in real-time, the point browser is refreshed old data will disappear.

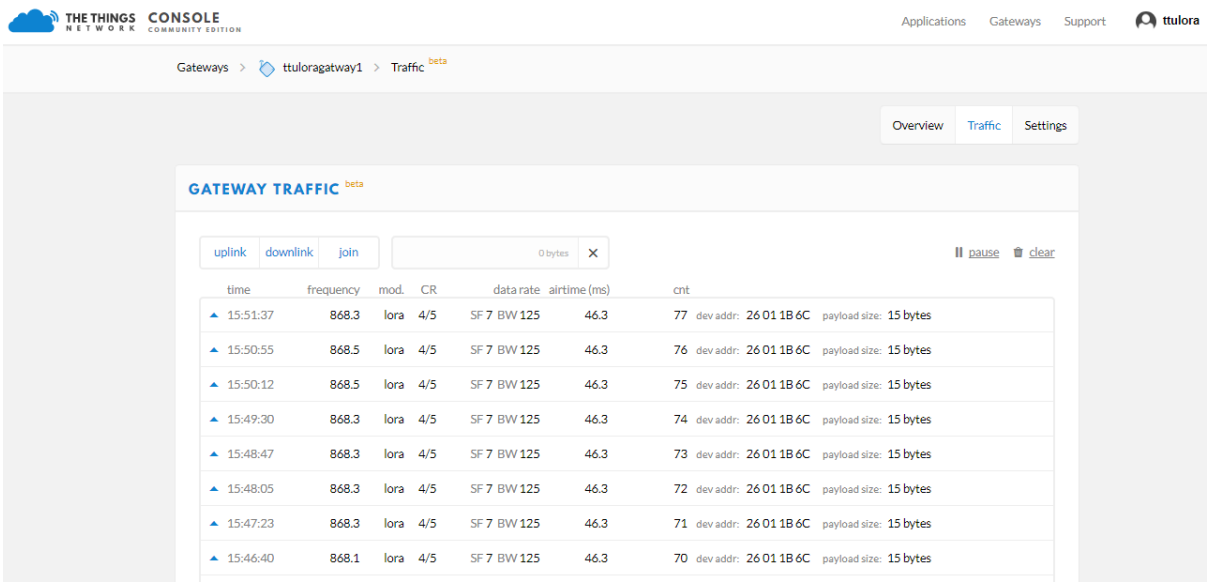

Figure 6: TTN Gateway traffic

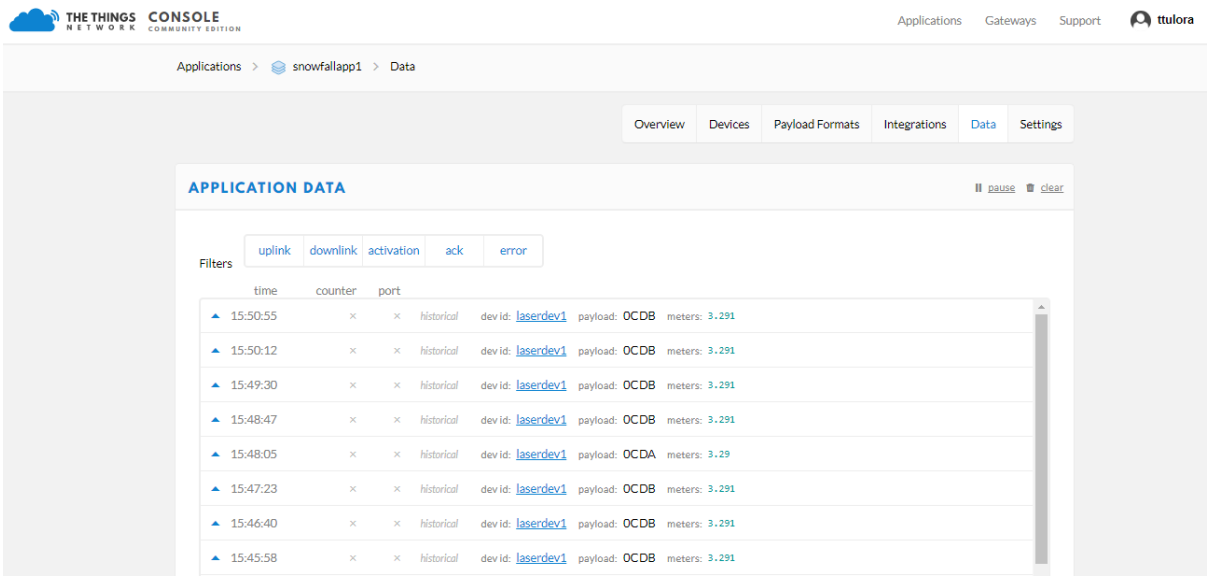

Figure 7: TTN Application data for registered device

6. Future Work

Pull data from TTN and represent in graphical format. As per my knowledge there are two ways to achieve this thing:

1) Install Node red on local system and pull data from TTN.

2) Use IBM Bluemix and pull data from TTN.

Implement same project on Sigfox environment. Sigfox environment is more reliable as it's provide OTAA (Over the air authentication) mode connection and have its own cloud service.

7. Conclusion

This project helps me to understand LoRa technology and tools available for developing lora application. Using this environment any LoRa based product can be designed.

LoRa has more payload capacity compare to Sigfox but Sigfox environment is more reliable in terms of packet transmission.

8. Reference

[1] https://github.com/johnmcdnz/LoPy/blob/master/Wifi%20config.md

[2] https://docs.pycom.io/chapter/tutorials/all/wlan.html

[3] https://www.thethingsnetwork.org/docs/gateways/multitech/aep.html

[4] http://help.ubidots.com/connect-your-devices/multiconnect-conduit-aep-lorawan-packet-forwarder

[5]https://github.com/TheThingsNetwork/packet_forwarder/blob/master/docs/INSTALL_INSTRUCTIONS/MULTITECH.md

[6] https://www.thethingsnetwork.org/docs/gateways/multitech/aep.html