Universität Bremen

Fachbereich 01 – Physik/Electrotechnik

Lora Network and Secure Communication for Digital Twins

Institute for Microsensors, Actuators and Systems (IMSAS)

Submitted by: Md Selim Hossain

Matriculation No: 3202203

First Examiner: Prof. Dr.-Ing. Walter Lang

Second Examiner: Dr.-Ing. Reiner Jedermann

Supervisor: Dr.-Ing. Reiner Jedermann

IMSAS

Institut für Mikrosensoren, -aktoren und -systeme

Declaration of Independence

I hereby confirm that I have written this thesis independently and have not used any resources other than those specified. The part of the work, which is taken from the wording or the meaning of other works, has been identified and the source has been indicated.

Table of Contents

List of Figures

List of Tables

Abstract

As the Internet of Things (IoT) continues its exponential growth, the demand for robust and efficient communication technologies in IoT environments becomes increasingly pressing. This project presents a comprehensive performance evaluation of LoRaWAN, a Low Power Wide Area Network (LPWAN) technology, within an indoor setting. The study was conducted within the confines of the NW1 building at the University of Bremen, focusing on assessing key performance parameters, including Signal to Noise Ratio (SNR), Received Signal Strength Indication (RSSI), and Packet loss.

Unlike many previous studies that primarily examined LoRaWAN's performance in outdoor scenarios, indoor environments have received limited attention despite their relevance to many IoT applications. The analysis includes a detailed investigation into the attenuation of RSSI in relation to varying distances, revealing a wide range of signal strengths ranging from -43 dBm to -113 dBm within the indoor environment. Additionally, the study extended to test the connectivity range of LoRaWAN over distances ranging from 10 to 90 meters, demonstrating that the LoRa Gateway can establish communication with end nodes at distances of up to 1.5 kilometres.

A significant observation was the inverse relationship between distance and RSSI value, accompanied by instances of packet loss as distances increased. These findings emphasize the importance of considering signal strength and distance when designing and deploying LoRaWAN networks for indoor IoT applications.

In summary, this project contributes valuable insights into the performance and suitability of LoRaWAN for indoor IoT deployments. The results presented here are instrumental in guiding network planners and researchers in optimizing IoT connectivity solutions in indoor environments, ultimately advancing the field of IoT communication technologies.

1 Introduction

Lora network is a technology that enables secure communication for digital twins, which are virtual replicas of physical objects or systems. Digital twins are used in various industries, including manufacturing, healthcare, and transportation, to monitor and control real-world assets remotely[1].

The Lora network ensures that the communication between digital twins and their physical counterparts is reliable and secure. It uses low-power, long-range wireless technology to transmit data over long distances, making it ideal for applications where devices are spread out across a wide area.

One of the key advantages of Lora network is its ability to provide end-to-end security[2]. This means that the data transmitted between digital twins and their physical counterparts is encrypted and can only be accessed by authorized parties.

RSSI, which stands for Received Signal Strength Indicator, is a critical metric in communication systems. It measures the strength of a received signal and provides valuable information about the quality and reliability of a wireless connection.

One of the key reasons why RSSI is important in communication is its ability to determine signal strength. By measuring the power level of a received signal, RSSI helps determine how well a device is able to receive and interpret data. This information is useful in wireless networks, where signal strength can fluctuate due to various factors such as distance, obstacles, or interference. By analyzing RSSI values, it is possible to assess the overall quality of a wireless connection[3]. This can help identify potential issues such as weak signals or high levels of interference, allowing for timely troubleshooting and optimization.

In addition to security, Lora network also offers scalability and flexibility. It can support a large number of devices simultaneously, making it suitable for applications with thousands or even millions of sensors. Furthermore, the network can be easily deployed and managed, allowing organizations to quickly set up their digital twin infrastructure.

The use of Lora network for secure communication in digital twins has numerous benefits. It allows organizations to monitor and control their assets remotely, reducing the need for physical presence and enabling cost savings. It also enables predictive maintenance, as data from digital twins can be used to identify potential issues before they occur.

2 Theoretical Overview

Lora, also known as Long Range or Low Power Wide Area Network (LPWAN), is an advanced wireless communication technology that has gained popularity in recent years. It is specifically designed for the Internet of Things (IoT) applications, where devices need to transmit data over long distances while consuming minimal power.

Lora uses spread spectrum modulation techniques to transmit data over the airwaves. It utilizes unlicensed frequency bands, such as the 868 MHz band in Europe and the 915 MHz band in North America. The use of unlicensed bands means that anyone can use Lora technology without needing to obtain a license.

The architecture of a Lora network consists of three main components: the end nodes, gateways, and network servers[4], [5]. The end nodes are the devices that collect and transmit data. They can be sensors, meters, or any other IoT device. These end nodes communicate with the gateways, which act as a bridge between the end nodes and the network servers.

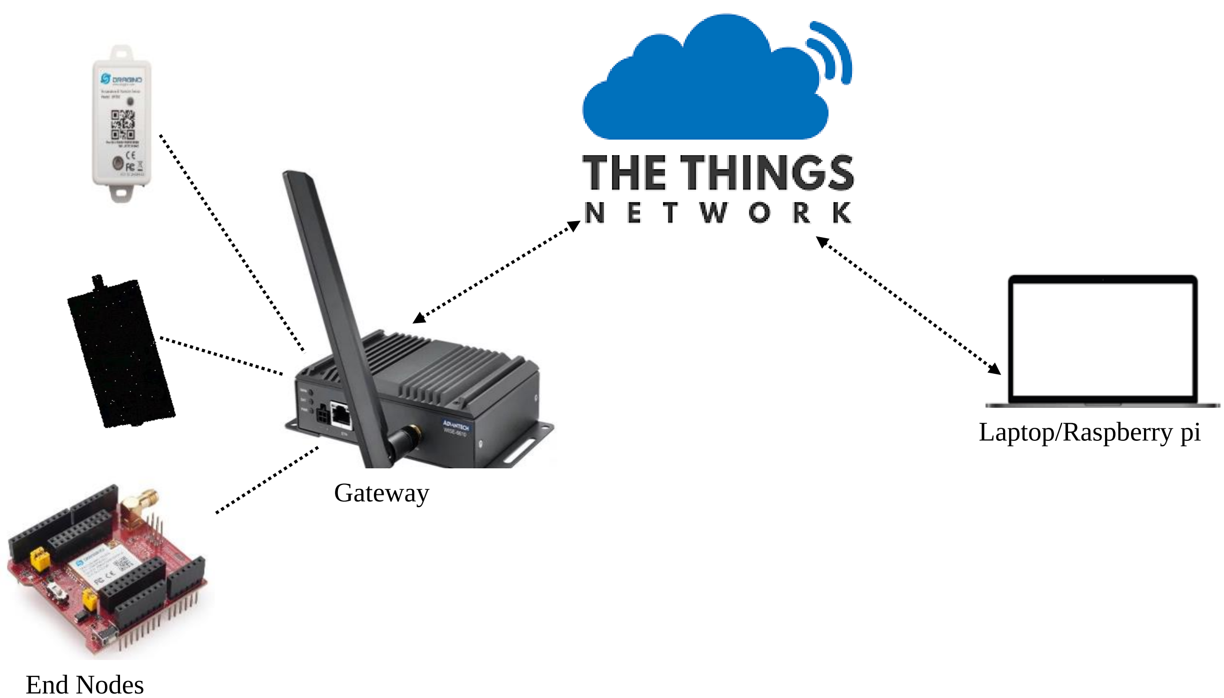

Figure 1 LoRaWAN Network Architecture.

One of the notable characteristics of Lora is its ability to provide long-range communication. It can cover distances of up to several kilometers in urban areas and even more in rural areas. This makes it ideal for applications such as smart cities, agriculture, and industrial monitoring, where devices are often spread out over a large area.

Another important aspect of Lora is its low power consumption. The end nodes in a Lora network can operate on battery power for years without needing to be recharged or replaced. This makes it cost-effective and convenient for deploying large-scale IoT networks.

When an end device transmits an uplink message to the LoRa gateway, it includes the RSSI (Received Signal Strength Indicator) value, which is obtained using an AT command on the end device. This RSSI value is then included as part of the transmitted message. The RSSI value indicates the power level of the received signal, usually measured in dBm (decibel-milliwatts). A higher RSSI value indicates a stronger signal, while a lower value suggests a weaker signal.

The RSSI value can be used to assess the distance between the end device and the gateway. As signals weaken with distance, a lower RSSI value can indicate that the end device is farther away from the gateway[6]. This information is valuable for determining coverage areas and optimizing network performance.

In addition to distance estimation, RSSI can also be used for signal quality assessment. By analyzing variations in RSSI values over time, network operators can identify areas with poor signal strength or interference issues. This allows them to take corrective measures such as adding additional gateways or optimizing antenna placement.

The combination of Lora with digital twin technology opens up new possibilities for various industries. A digital twin is a virtual representation of a physical object or system. It enables real-time monitoring and analysis of data from the physical counterpart, allowing for predictive maintenance, optimization, and simulation.

Overall, Lora with digital twin technology has the potential to revolutionize industries by providing real-time visibility and control over physical assets.

2.1 Related Work:

This paper[4] offers an extensive examination of LoRaWAN architecture, protocol, and technologies. It encompasses a detailed discussion of LoRaWAN's architecture components, including network servers, gateways, and end nodes, along with an exploration of the underlying physical layer technology involving LoRa modulation. Additionally, the paper delves into essential aspects like the mathematical model for frame air time calculations, MAC commands, energy efficiency considerations, security measures, and activation methods within the context of LoRaWAN. The paper also highlights the efficiency, performance, and limitations of LoRaWAN, as well as future research directions.

Eyuel D. Ayele[6] proposed, In the context of Low-Power Wide-Area Network (LPWAN) protocols for Internet of Things (IoT) applications, this study focuses on the analysis of LoRa wireless technology. The research provides a comprehensive overview of LoRa modulation and its network architecture while delving into key physical layer parameters such as spreading factor, bit rate, and coding rate. A practical LoRa radio network was constructed and prototyped to assess its network performance. The investigation of LoRa radio Received Signal Strength Indicator (RSSI) values revealed LoRa's resilience against multi-path and signal fading, particularly at higher spreading factors, attributed to its broadband chirp pulses and heightened sensitivity. At closer distances to the gateway, a low spreading factor resulted in higher RSSI, while increased spreading factors reduced packet loss but sacrificed effective bit rates, making them less suitable for high-throughput IoT applications. In contrast, at greater distances from the gateway, interferences were notably high, necessitating end-devices to communicate using higher spreading factors

Kwasme, Hussein[7] proposed, In this paper the researchers studies into an in-depth exploration of LoRaWAN RSSI-based localization, analyzing its accuracy, potential limitations, and future prospects. Furthermore, the integration of software-defined radios (SDR) is introduced to characterize path-loss effects. The experimental results highlight the considerable difficulty presented by RSSI variance caused by LoRaWAN's frequency hopping feature, which can have a detrimental impact on localization performance.The paper concludes by proposing innovative solutions aimed at mitigating this issue and ultimately enhancing the overall performance of RSSI-based LoRaWAN localization.

According to the paper[8], RSSI (Received Signal Strength Indicator) is used as a measurement parameter in LoRaWAN (Long Range Wide Area Network) to evaluate the transmission performance. The paper discusses various measurement scenarios and their corresponding RSSI values.In an outdoor measurement-based study, the authors evaluate the system performance under different payload length, bandwidth, spreading factor, and modulation schemes. They use RSSI and PER (Packet Error Rate) to estimate the system's performance. The study takes into account the Fresnel zone and compares the results with a different payload length and FSK modulation scheme.

This paper[9] explores the feasibility of using Dense Indoor LoRaWAN for passively sensing human presence. The researchers developed a Dense Indoor Sensor Network (DISN) with 390 sensor nodes and three gateways and evaluated its performance over six months. The study collected a large dataset of over 14 million transmissions and investigated factors affecting network performance, such as signal strength, distance, and the number of floors. The results showed that signal strength decreased with distance and the number of floors, indicating lower coverage indoors. The researchers recommend placing multiple gateways every 30 meters and 5 floors to ensure good signal coverage. The study also discusses the use of DISNs for passive sensing and visualization of human presence using a Digital Twin. Overall, the paper provides insights into the performance and limitations of LoRaWAN in indoor environments and highlights potential applications for analyzing human presence in buildings

3 Methodology

To begin the process, the ThingsNetwork account was created and access to their console. The console provides a user-friendly interface for managing LoraWAN devices and network settings. Within the console, the user can create an application and register their LA66 LoRaWAN, RN2483, and LHT65 module as a device within that application.

When using a LoRaWAN and RN2483, the parameters need to be set manually before the device can send data. How to set these parameters completely depends on the type of chip that is used in the device. Most of the times this information can be found in the documentation.

The next step is to configure the LA66 LoRaWAN, RN2483, and LHT65 module with the appropriate settings. This includes providing the necessary device address, network session key, and application session key. These settings can be obtained from the ThingsNetwork console and should be properly entered into the LA66 LoRaWAN, RN2483, and LHT65 module configuration. Once the modules are configured, it can be connected to the ThingsNetwork by selecting the appropriate frequency plan and joining method.

3.1 Uplink Transmission

In LoRaWAN, uplink transmission occurs in two steps: device-to-gateway and gateway-to-network server. Firstly, the IoT device sends its encoded data to the nearest gateway using the LoRa modulation scheme. This modulation allows the signal to travel long distances while consuming minimal power. The gateway then receives the signal and forwards it to the network server.

3.1.1 Uplink Transmission with LA66 LoRaWAN Module

In this project,the development board from STMicroelectronics, namely the NUCLEO-L031K6 board represents the end node. This board is especially notable for its low-power MCU. This microcontroller was connected with LA66 LoRaWAN and SHT31 temperature and humidity sensor. The MCU reads the sensor data every 10 seconds, converts it to hex values, and then transmits it through the LA66 LoRaWAN module.

Figure 1 represents the overall work flow of the uplink process. The workflow chart of uplink to ThingsNetwork outlines the process of sending data from a device to the ThingsNetwork platform.

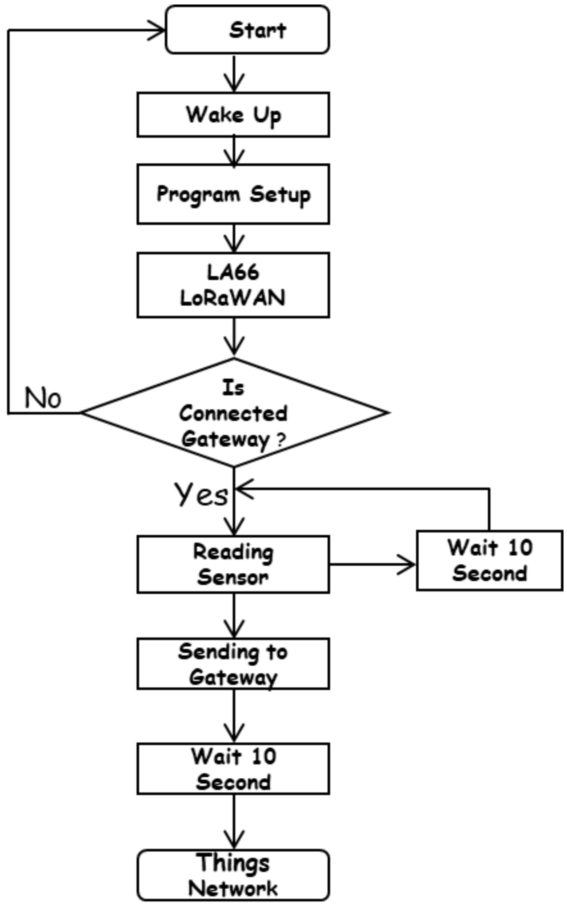

Figure 2 Flow Diagram of Uplink Transmission with LA66 LoRaWAN Module.

3.1.2 Flow Diagram of LA66 LoRaWAN Module

The working procedure of the flow diagram is given below.

1. **Start**: This is the beginning of the process or program.
2. **Wake Up**: This step represents the waking up or initialization phase of the system or device.
3. **Program Setup**: In this stage, the program or system is configured and set up for operation.
4. **LA66 LoRaWAN:** The flowchart proceeds to the "LA66 LoRaWAN" step, which signify the activation or initialization of a specific LoRaWAN communication module or component.

5. **Connected to Gateway:** After setting up the LoRaWAN module, the system establishes a connection to a LoRaWAN gateway. The gateway acts as a bridge between the device and the network.

6. **Reading Sensor:** This phase involves collecting temperature and humidity data from the SHT-31 sensor.

7. **Sending Data to Gateway:** The collected sensor data is transmitted or sent to the LoRaWAN gateway for further processing.

8. **Wait 10 seconds:** wait 10 seconds for next data transmission.

9. **Send Data to The Things Network**: Finally, the processed data is sent to The Things Network, which is a platform for managing LoRaWAN devices and data.

3.1.3 Experimental Uplink Setup with LoRaWAN Module

The figure 3 and 4 show the flow diagram of the experimental uplink setup. In this project, we will take a look at a STM32 Nucleo boards from STMicroelectrocis & send temperature and humidity real time data to The Things Network Server with the help of LoRaWAN Gateway. In the uplink transmission, the SHT-31 temperature and humidity sensors, Nucleo mictrocontroller, LA66 LoRaWAN Module, LoRawan Gateway equipment are present. The data from the sensors are read

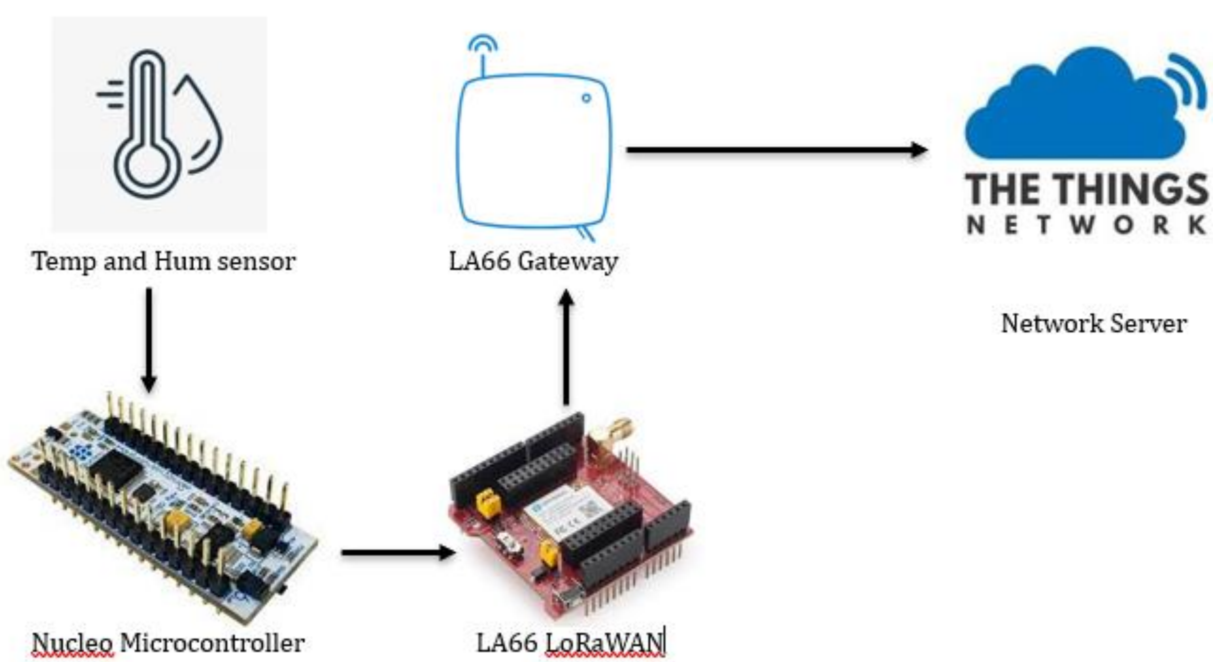

Figure 3 Uplink Setup with LoRaWAN Module.

by the Nucleo microcontroller and passed it through LA66 LoRaWAN module and finally, send the sensor data to the gateway (uplinks), and the gateway pass it on to the network server (The ThingsNetwork), which, in turn, passes it on to the application server as necessary.

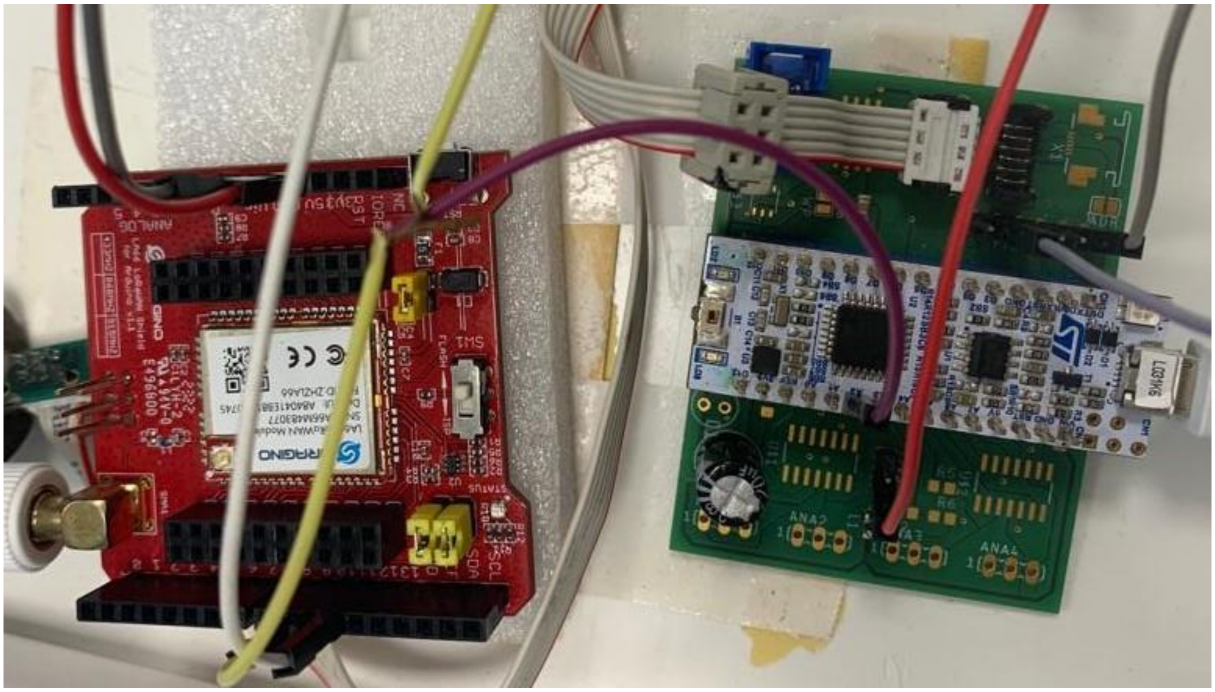

Figure 4 Experimental Setup for Uplink Transmission with LoRaWAN Module.

3.1.4 Configure the LA66 LoRaWAN Module:

To set up the LA66 LoRaWAN module, it's essential to follow the steps below[10]:

1. Connect the LA66 LoRaWAN module to a PC with the help of Serial wire. The device will create a new serial device (COMx on Windows and /dev/ttyACMx on Linux).
2. Open a terminal session to the serial device/port with the following parameters: bitrate: 9600, 8 bits, no parity, 1 stop bit, no flow control.
3. Commands need to be terminated with <CR><LF>.

4. Set the Device address.

```
AT+DADDR= < 8-digit value from the ttn>
```

5. Set the AppKey:

   `AT+APPKEY= < 32-digit value from the ttn >`

6. Set the AppEUI:

   `AT+APPEUI= <16-digit value from the ttn>`

7. Set the NwksKey:

   `AT+NWKSKEY= < 32-digit value from the ttn >`

8. Set the AppsKey:

   `AT+APPSKEY= < 32-digit value from the ttn >`

3.1.5 Uplink Transmission with RN2483 Module

In this project, the RN2483 is used as an end node which is responsible for uplink transmissions. The RN2483 module serves as the endpoint device that sends data to the LoRaWAN gateway. Configuration is a necessary prerequisite before transmitting data to the gateway.

Figure 5 displays the flowchart illustrating the uplink transmission process for the RN2483.

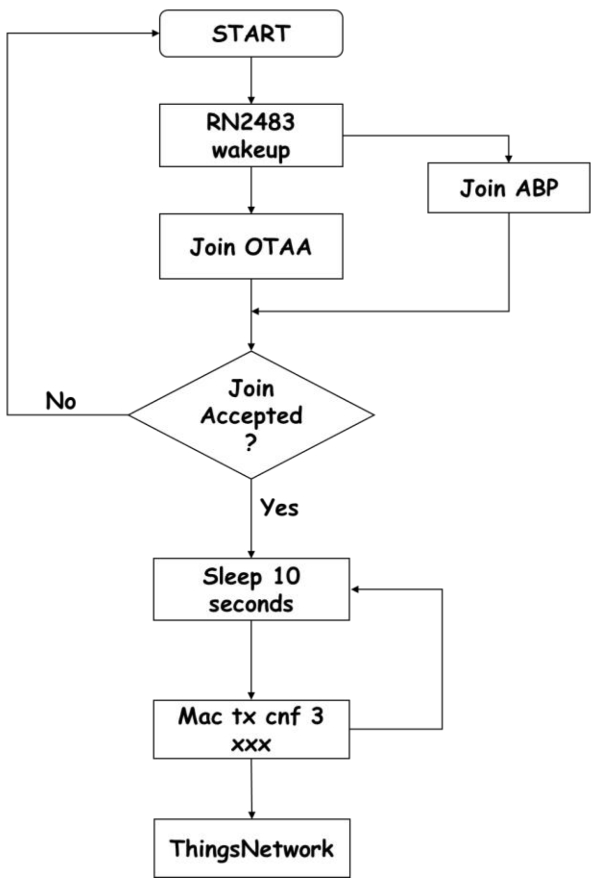

Figure 5 Flow Diagram of Uplink Transmission with RN2483 Module.

3.1.6 Flow Diagram with RN2483

The working procedure of the flow chart is discussed below.

1.**Start:** The flow chart begins at the "Start" point, indicating the initiation of a process.

2. **RN2483 Wake Up**: This step signifies that the RN2483 module is being powered on or awakened from a low-power state.

3. **Join OTAA/ Join ABP**: In this step, the RN2483 module initiates the process of joining a LoRaWAN network using the Over-the-Air Activation (OTAA) method or Activation by Personalization (ABP) . This involves exchanging secure credentials with a LoRaWAN gateway to establish a connection.

4. **Connected to Gateway**: Once the RN2483 module successfully joins the LoRaWAN network, it establishes a connection with a LoRaWAN gateway. This gateway acts as a bridge between the module and the network server.

5. **Wait 10 Seconds**: This step introduces a 10-second delay, during which the RN2483 module may be waiting for synchronization, performing initialization, or simply pausing for a specific purpose.

6. **Transmit to ThingsNetwork**: After the 10-second delay, the RN2483 module proceeds to transmit data to The Things Network (TTN).

3.1.7 Experimental Uplink Setup with RN2483 Module

The laptop or Raspberry pi is connected to the RN2483 module via a serial communication link. This connection allows data to be exchanged between the laptop/Raspberry pi and the module. Ensure that the serial communication parameters (e.g., baud rate, data bits, parity, stop bits) on both the laptop and the RN2483 module are configured to match for proper serial data exchange. The RN2483 module, equipped with built-in LoRa technology, is responsible to transmit data wirelessly to the LoRa gateway. The LoRaWAN gateway acts as a bridge, receiving data from the RN2483 module. The gateway forwards the data to The Things Network for further processing. The Things Network receives the data from the gateway and processes it based on the configured network settings.

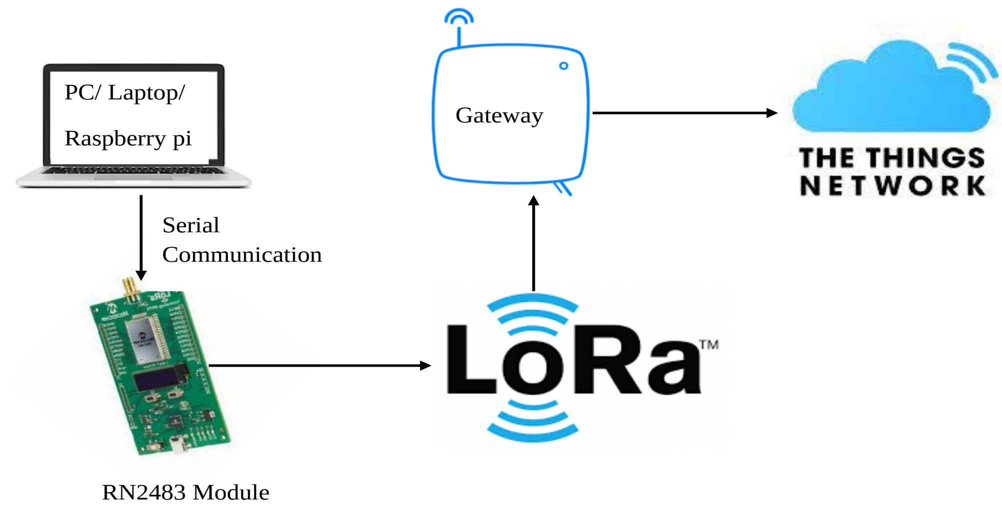

Figure 6 Uplink Setup with RN2483 Module.

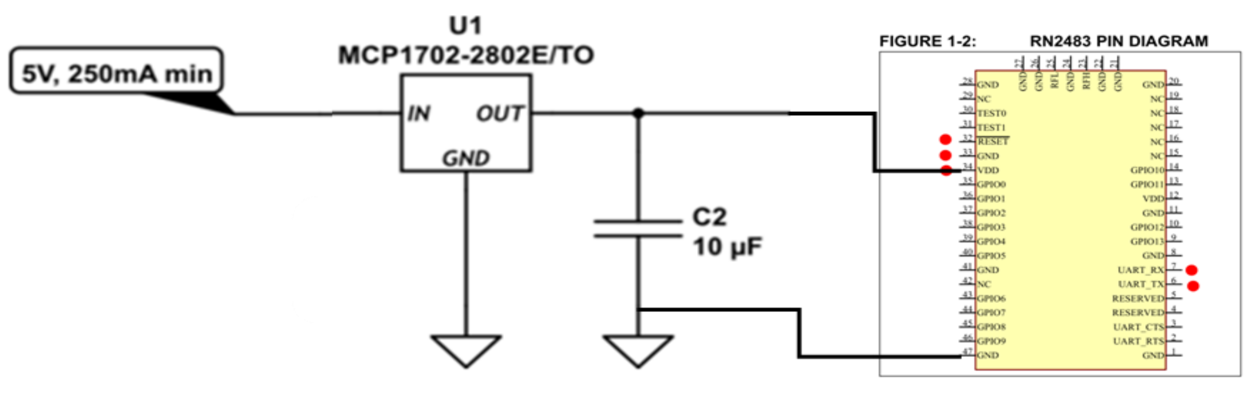

Figure 7 Schematic Setup for Uplink Transmission with RN2483 Module.

3.1.8 Configure the RN2483:

Configuration is achieved through an ASCII command interface using UART. The RN2483 board contains with a USB port that is integrated USB-to-serial functionality. Similarly, The RN2483 (Naked) comes with Tx and Rx pins designed for serial communication, and these pins are linked to the USB port, enabling convenient data exchange between the module and external devices.

In order to configure the RN2483 module, it is necessary to follow the steps outlined below[11]:

1. Connect the module to a PC. The device will create a new serial device (COMx on Windows and /dev/ttyACMx on Linux).
2. Open a terminal session to the serial device/port with the following parameters: bitrate: 57600, 8 bits, no parity, 1 stop bit, no flow control.
3. Commands need to be terminated with <CR><LF>.

4. Get the hwEUI (physical address) of the module.

```
sys get hweui
0004A30B001B4D55
```

5. Set the AppKey:

```
mac set appkey 00112233445566778899AABBCCDDEEFF
ok
```

6. Set the AppEUI:

```
mac set appeui FEDCBA9876543210
ok
```

7. Save the settings in the device:

```
mac save
ok
```

3.1.9 Uplink Transmission with LHT65:

The LHT65 module serves as an end node for uplink transmission, requiring registration with the Things Network for data transmission. Once activated, it automatically sends data to the gateway every 10 seconds. If you need to modify its configuration, please follow the configuration steps provided below.

3.1.10 Experimental Uplink Setup with LHT65 Module:

In this project, The LHT65 module, configured as an end node. No additional connections to laptops or microcontrollers are required for this setup. Register the LHT65 module with The Things Network by providing the necessary device information, including Device EUI, Application EUI, and App Key. once powered on and registered, will autonomously transmit data wirelessly to The Things Network at predefined intervals, in this case, every 2 minutes.

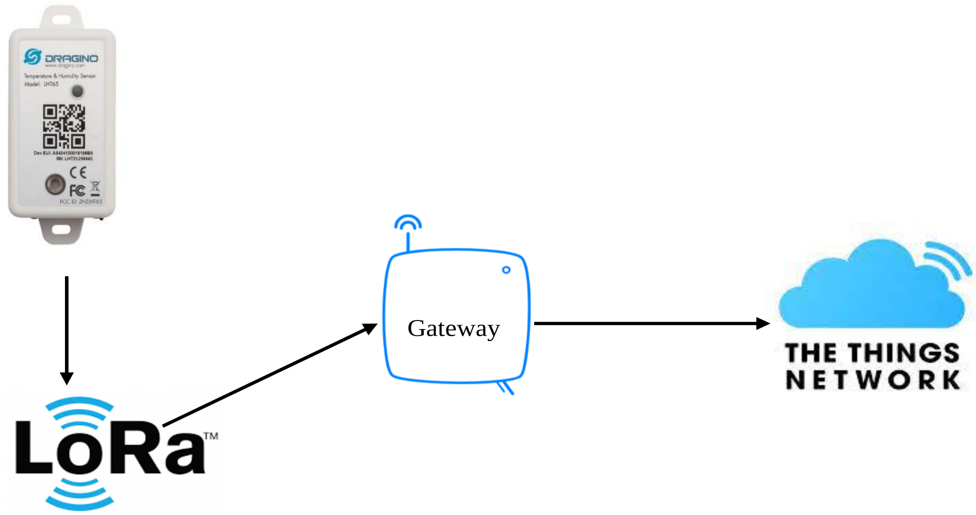

Figure 8 Uplink Setup with LHT-65 Module.

3.1.11 Configure the LHT65

The LHT65 modules come with preconfigured keys including the Device EUI, Application EUI, and Application Key. Complete the registration of the LHT65 module on The Things Network (TTN) by using the Device EUI, Application EUI, and Application Key. If the keys ever need to be updated or modified for any reason, the steps outlined below should be followed[12].

The LHT65 device supports an AT Command set. Users can use a USB to TTL adapter along with the Program Cable to establish a connection with the LHT65 device for utilizing AT commands[13].

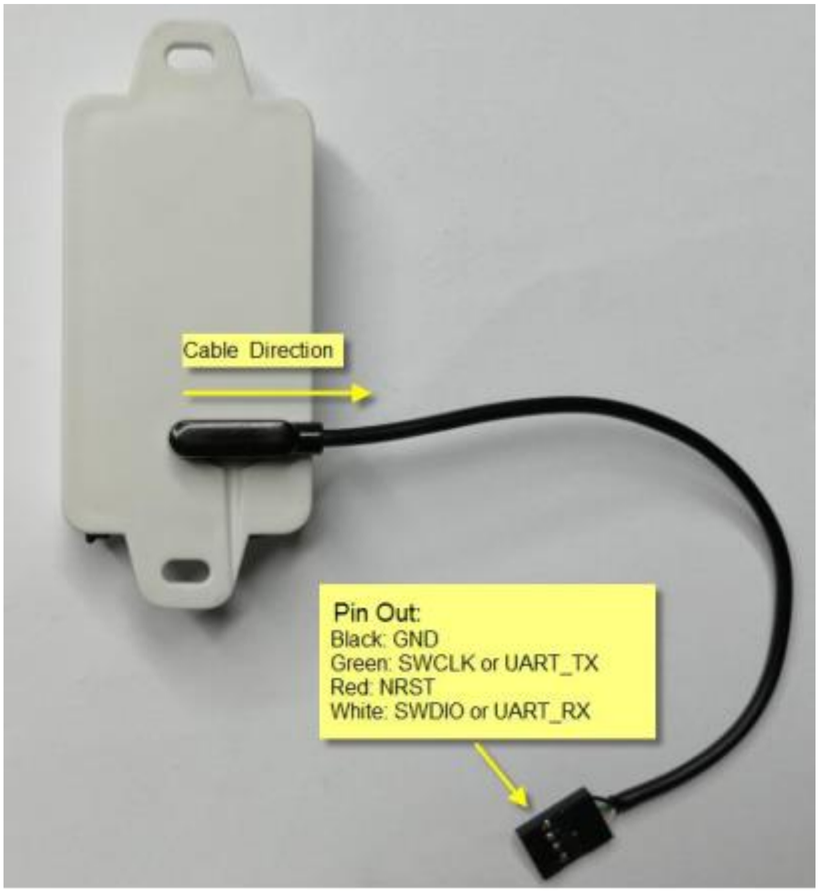

Figure 9 : LHT 65 with program cable[13].

Since the USB to TTL adapter typically uses only the TX (Transmit) and RX (Receive) pins for serial communication, it often doesn't require a voltage regulator. However, it's essential to ensure that the voltage levels of the USB to TTL adapter match the requirements to avoid potential damage.

Connection:

USB to TTL GND <---> Dupont black pin

USB to TTL RXD <---> Dupont green pin

USB to TTL TXD <---> Dupont white pin

By default, the AT commands are deactivated, and users must input the password (default: 123456) to enable them. It's important to note that there is a 5-minute timeout for entering AT commands. After this 5-minute period, users will need to input the password again to reactivate the AT command functionality.

1. Connect the LHT65 module to a PC with the help of Serial wire. The device will create a new serial device (COMx on Windows and /dev/ttyACMx on Linux).
2. Open a terminal session to the serial device/port with the following parameters: bitrate: 9600, 8 bits, no parity, 1 stop bit, no flow control.
3. Commands need to be terminated with <CR><LF>.

4. Set the Device address.

   `AT+DADDR= < 8-digit value from the ttn>`

5. Set the AppKey:

   `AT+APPKEY= < 32-digit value from the ttn >`

6. Set the AppEUI:

   `AT+APPEUI= <16-digit value from the ttn>`

7. Set the NwksKey:

   `AT+NWKSKEY= < 32-digit value from the ttn >`

8. Set the AppsKey:

   `AT+APPSKEY= < 32-digit value from the ttn >`

The default data transmission interval is set at 30 minutes. To modify the transmission interval, utilize the following command format:

`AT+TDC = 10 * 1000`

In this project, the data transmission interval is configured to occur every 10 seconds.

3.1.12 Uplink Message

After successfully connecting to ThingsNetwork, The ThingsNewtwok server receives the uplink messages from the end devices. After receiving the uplink message, it becomes necessary to apply a filtering process to the message to meet our specific criteria and needs. This filtering step is important as it allows us to extract relevant information and filtering out irrelevant information.

From the uplink message, it is possible to calculate the RSSI values (Received Signal Strength Indicator) for incoming messages. RSSI provides information about the signal strength of a received message, which can be valuable for analyzing network performance and troubleshooting connectivity issues.

3.2 Downlink Transmission

Downlink transmission plays a fundamental role in LoRaWAN as it enables bidirectional communication, allowing for commands, acknowledgments, and updates to be sent from the server to the devices. This is especially important in applications where real-time data exchange is required, such as in smart cities, agriculture, and industrial monitoring.

3.2.1 Downlink Transmission with LA66 LoRaWAN Module

The downlink feature allows for real-time control and monitoring of devices, enabling users to remotely manage their IoT devices and respond quickly to any issues or changes. The ThingsNetwork's downlink feature plays a crucial role in enabling bi-directional communication between IoT devices and applications. The flow chart below illustrates the downlink process for this project.

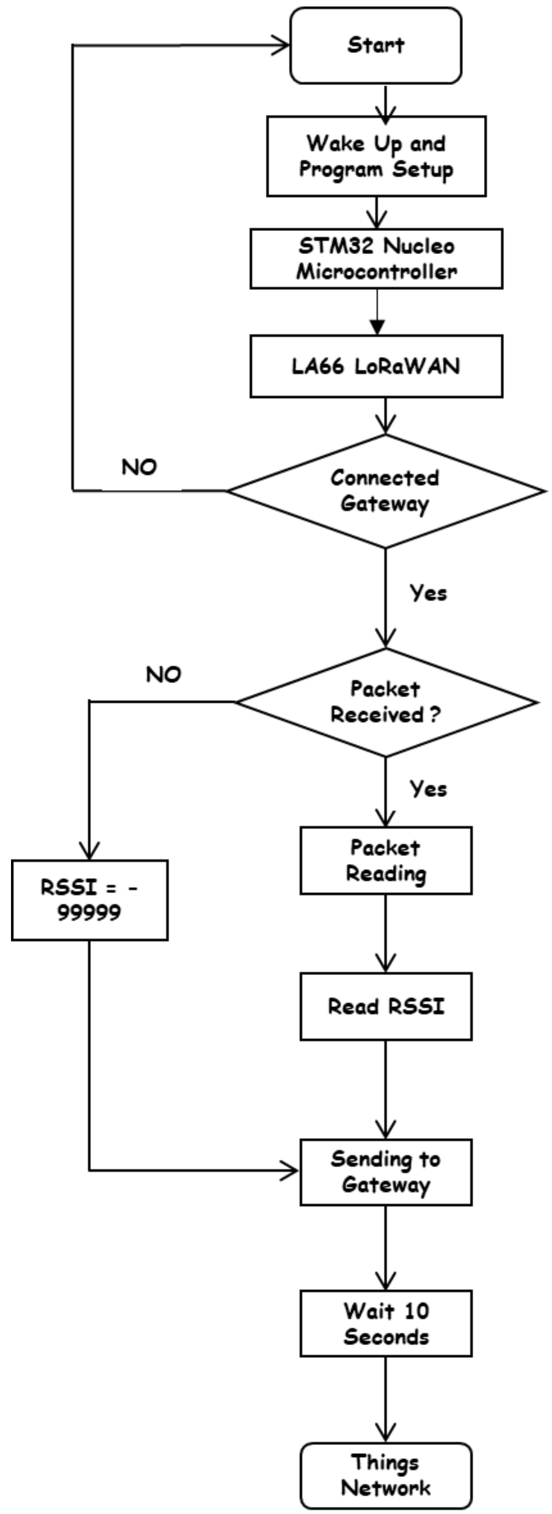

Figure 10 Flow Diagram of Downlink Transmission.

3.2.2 Flow Diagram of LA66 LoRaWAN Module

1. **Start:** The flow chart begins here, marking the initiation of the process.
2. **Wake Up and Program Setup:** This step involves the initialization of the system and any necessary program setup procedures.
3. **STM32 Microcontroller:** The STM32 microcontroller plays a central role in the process, managing various tasks and coordinating communication.
4. **LA66 LoRaWAN:** The LA66 LoRaWAN module is responsible for LoRa communication, allowing the device to transmit and receive data over long distances.
5. **Connected to Gateway:** The system establishes a connection with the LoRaWAN gateway, which acts as a bridge between the device and the network.
6. **Packet Received:** This step indicates that the device has successfully received a packet of data from the network or another device.
7. **Packet Reading:** The received packet is processed in this step, where the data is extracted and prepared for further analysis or action.
8. **Read RSSI:** The Received Signal Strength Indicator (RSSI) is read from the received packet, providing insights into signal quality.
9. **Sending to Gateway:** After processing the data, the system sends its response or data back to the LoRaWAN gateway.
10. **Wait 10 Seconds:** A 10-second delay is introduced, potentially for synchronization or timing considerations within the system.
11. **The Things Network:** Finally, the data is transmitted to The Things Network, which serves as the central hub for managing and routing LoRaWAN data, enabling it to reach its intended destination or application.

3.2.3 Experimental Downlink Setup with LoRaWAN Module

The figure 10 shows the flow diagram of the experimental downlink setup. The downlink function is achieved through the use of gateways, which act as intermediaries between the devices and the applications.

To transmit downlink data to The Things Network (TTN) using Node-RED, first install Node-RED and access it through a web browser. Typically, Node-RED includes the necessary nodes for LoRaWAN integration. Ensure LoRaWAN device is connected to the gateway and configured to receive TTN downlink messages. Create a Node-RED flow to trigger downlink messages, define the payload format, and use available LoRaWAN nodes for TTN communication. For automated triggering, include time-based or input nodes to schedule downlink transmissions.First, the node-red is connected to the ThingsNetwork's APIs server using mqtt out node. To initiate a downlink, the Nucleo microcontroller first sends a message to the device via the network server. This message contains the necessary instructions for the device to carry out a specific action. The network server then forwards this message to the appropriate gateway, which in turn transmits it to the targeted Nucleo device. After receiving the downlink packets, the rssi value is extracted from the packets and saved in a file for further analysis.

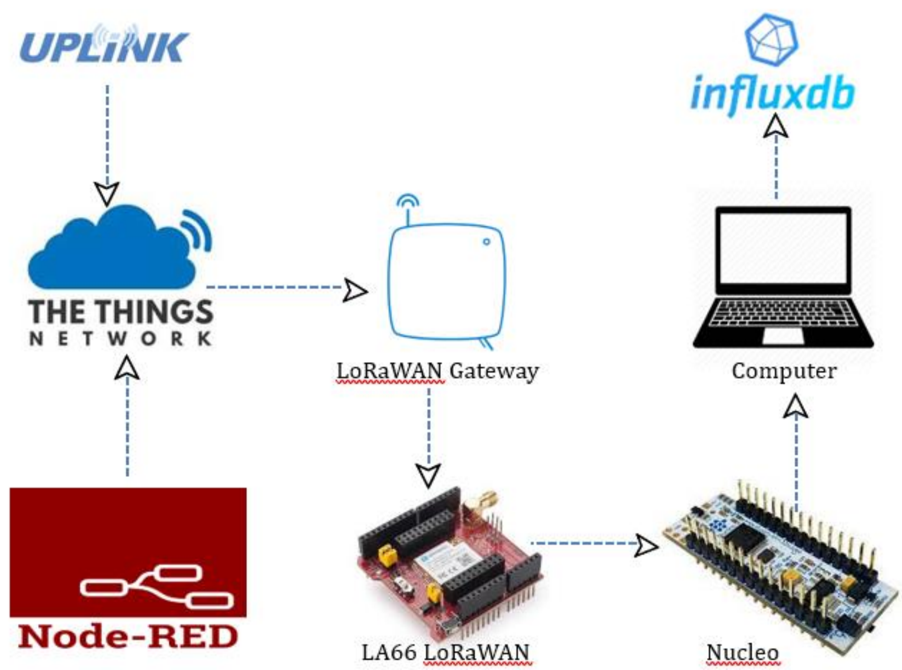

Figure 11 Overview of Downlink Transmission

3.2.4 Downlink Message

When a packet is received by the LA66 module, it means that data has been successfully transmitted from another device and received by the module. This module is equipped with the necessary hardware and firmware to decode and process the incoming packets of data. The received packets can contain various types of information, depending on the application and use case. In this project, the Received Signal Strength Indicator (RSSI) is extracted from the received packet. This functionality allows for the evaluation of signal strength and quality for received packets. The RSSI value is an important metric in determining the reliability and performance of wireless communication. By analyzing the RSSI value, users can assess the signal's strength and make informed decisions regarding network optimization and troubleshooting. The RSSI value is set to -99999 if the LA66 LoRaWAN module does not receive the packet.

3.3 Voltage Regulator

In this project, the LA66 LoRaWAN and RN2483 modules operate on a 3.3V power supply, whereas the output from a typical power bank or laptop USB port is 5V. To provide the required 3.3V power to these modules, a voltage conversion or regulation step is necessary[14]. This can be achieved using a voltage regulator or voltage converter module.

One commonly used component for this purpose is a low dropout voltage regulator (LDO) like the MCP1700. LDOs are designed to provide a stable output voltage even when the input voltage is very close to the desired output voltage, making them suitable for applications like this one.

To implement the voltage conversion from 5V to 3.3V, Connect the input of the LDO voltage regulator to the 5V source (from the power bank or laptop USB) and connect the output of the LDO to the LA66 LoRaWAN and RN2483 modules, ensuring they receive a stable 3.3V supply.

The circuit diagram and experimental setup are shown below

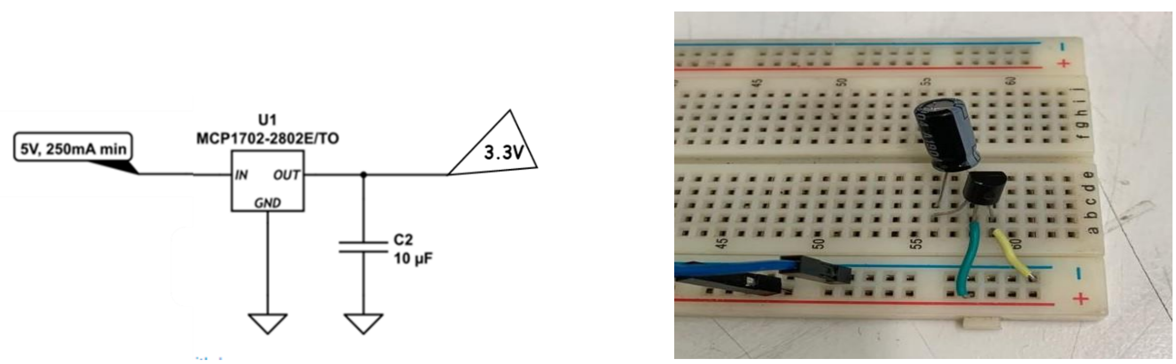

Figure 12 Schematic of Voltage Regulator with MCP1700

By using the LDO voltage regulator in this way, LA66 LoRaWAN and RN2483 modules receive a consistent and appropriate power supply voltage.

4 System Overview

In this project, the LoRaWAN platform is designed with a carefully crafted structure that ensures optimal performance and security. The platform consists of several key components that work together seamlessly to provide a reliable and scalable solution.This LoRaWAN platform consists of several layer such as device layer, network layer, information layer and Analysis layer[15].

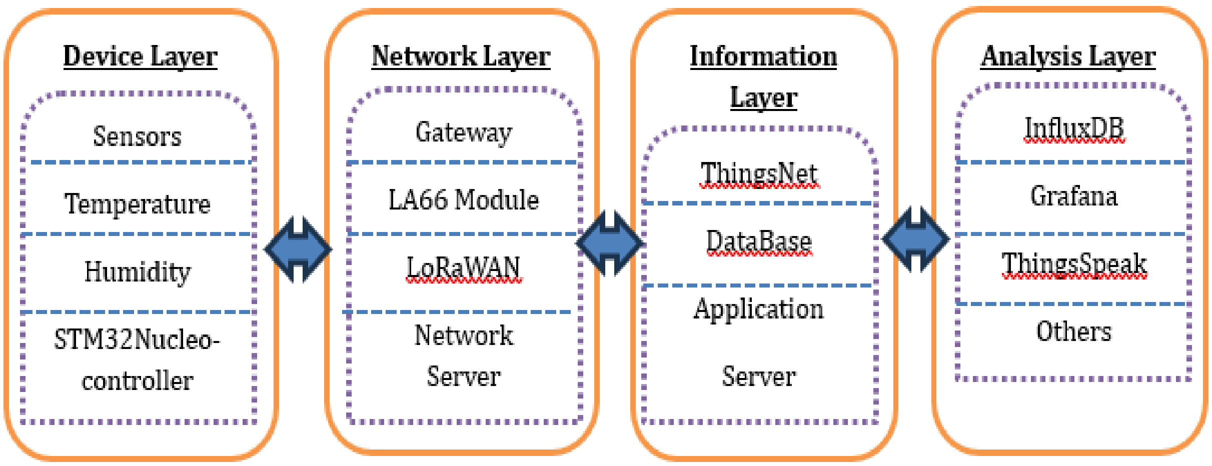

Figure 13 Overview of LoRaWAN platform Layers

- **Device Layer**

  The device layer is a critical component of the Internet of Things (IoT) ecosystem, serving as the interface between physical devices and the rest of the IoT architecture. It includes STM32 Microprocessor, sensors, actuators, and other hardware components that enable devices to collect data and interact with their environment. The microprocessor should be32-bit or higher processors such as STM32 or ARM derivatives. The device layer must be designed to handle a variety of challenges, including low-power consumption, limited processing resources, and diverse communication protocols.

- **Network Layer**

  This layer consists of the following connection as shown on the figure 12, including the Gateway, LA66 LoRaWAN module and the Network Server.

  The network layer in LoRaWAN is responsible for managing the communication between end devices and gateways, as well as the routing of data packets. This layer uses a star-of-stars topology, where gateways act as intermediaries between end devices and the network server. The network server is responsible for managing the entire network and ensuring that data packets are delivered to their intended destination. LoRaWAN also uses adaptive data rate (ADR) to optimize the transmission rate based on the signal quality, which helps to conserve battery power in end devices. Another important feature of LoRaWAN is its use of encryption and authentication mechanisms to ensure the security of data transmission. Overall, the network layer in LoRaWAN plays a crucial role in enabling reliable and secure communication between IoT devices and the Gateways.

- **Information Layer**

  This layer acquires data from sensors such as temperature sensor, humidity sensor, timestamps, and other sensor data depending on the application.

  The information layer in LoRaWAN is an integral part of this protocol. It is responsible for managing the data transmission between IoT devices and the application server. This layer ensures that data is transmitted accurately and efficiently, minimizing the amount of power required by the IoT device.

  Overall, the information layer in LoRaWAN deals with the actual payload data, encoding, encryption, modulation, and related parameters necessary for the secure and efficient transmission of data between IoT devices and the network.

- **Analysis Layer**

  This layer consists of the following components as shown on the figure x, including the InfluxDB, Grafana, Thingsspeak, AWS IoT and Azure IoT. This layer utilizes all the information provided by the network and information layers.

  One of the key features of this layer is its ability to prioritize certain types of data, such as emergency messages or critical updates, over less important traffic. This helps to ensure that the most important information is delivered in a timely and efficient manner. Additionally, the analysis layer is designed to be flexible and adaptable, allowing it to respond to changing network conditions and traffic patterns.

4.1 Hardware

The following tools and components are used to make this project.

S.N.	COMPONENTS NAME	QUANTITY
1	SHT – 31 Sensor	1
2	STM32 Nucleo Microcontroller	1
3	LA66 LoRaWAN USB Module	3
4	LA66 LoRaWAN Normal Module	3
5	LoRaWAN Gateway	1
6	3.7V, 10000mAH Lithium-Ion power bank	1
7	Jumper Wires	5-6
8	Micro-USB Cable	1
9	RN2483 Module	6
10	LHT-65 Temp and Humidity Sensor	3
11	MCP1700 Voltage Regulator	6
12	Capacitor	6
13	USB-to-Serial Cable	6

Table 1 List of Hardware components.

4.1.1 SHT – 31 Sensor

The SHT31 sensor is a highly accurate and reliable sensor that is widely used in various industries and applications. It is known for its exceptional performance and advanced features.

Figure 14 SHT-31 Temperature and Humidity Sensor[16]

One of the notable characteristics of the SHT31 sensor is its exceptional level of precision. It provides precise measurements of temperature and humidity, ensuring that the data collected is reliable and trustworthy. Another notable feature of the SHT31 sensor is its fast response time. It can quickly detect changes in temperature and humidity, allowing for real-time monitoring and adjustments[17]. This makes it ideal for applications where rapid response is necessary, such as in climate control systems or environmental monitoring.

The SHT31 sensor also stands out for its low power consumption. It has been designed to operate efficiently, minimizing energy usage without compromising on performance. This makes it suitable for battery-powered devices or applications where power efficiency is a priority.

In terms of connectivity, The SHT-31 sensor communicates via a standard I2C (Inter-Integrated Circuit) digital interface, making it easy to integrate into microcontroller-based systems.

Overall, the SHT31 sensor is a top choice for professionals seeking accuracy, reliability, and advanced features in temperature and humidity sensing. Its exceptional performance, fast response time, low power consumption, wide operating range, and versatile connectivity options make it a valuable tool in a wide range of industries and applications.

4.1.2 STM32 Nucleo Microcontroller

The STM32 Nucleo is a series of microcontroller development boards created by STMicroelectronics[18]. The STM32 microcontroller core is a highly versatile and powerful system-on-chip (SoC) that is widely used in a variety of applications. It offers a range of features and capabilities that make it ideal for embedded systems development.

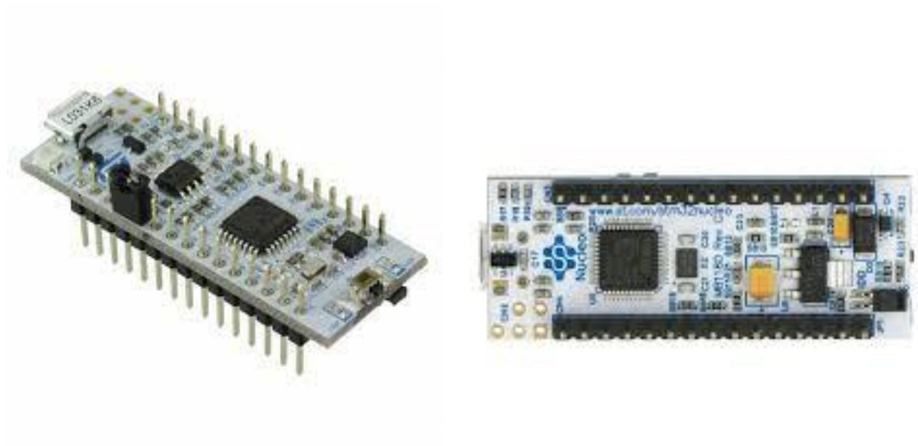

Figure 15 STM-32 Nucleo Microcontroller

One key aspect of the STM32 microcontroller core is its powerful processing capabilities. It employs an ARM Cortex-M processor, which provides excellent performance and energy efficiency. This allows for high-speed data processing and efficient power consumption, making it suitable for both low-power and high-performance applications.

Another important feature of the STM32 microcontroller core is its extensive set of peripherals. It offers a wide range of communication interfaces, including UART, SPI, I2C, USB, and Ethernet. This enables seamless connectivity with other devices and systems, making it highly versatile in various networking applications.

Furthermore, the STM32 microcontroller core comes with a comprehensive development ecosystem. This includes software development tools, such as integrated development environments (IDEs) and debugging tools, which facilitate easy code development and debugging.

In conclusion, the STM32 microcontroller core offers a wide range of features and capabilities that make it an excellent choice for embedded systems development. Its powerful processing capabilities, extensive peripheral options, memory support, and comprehensive development ecosystem make it highly versatile and efficient.

4.1.3 LA66 LoRaWAN Module

The LA66 LoRaWAN module is a cutting-edge technology that offers long-range, low-power wireless communication for various IoT applications. This module is designed to provide seamless connectivity and efficient data transmission over long distances, making it ideal for applications such as smart cities, agriculture, asset tracking, and industrial automation.

Figure 16 LA66 LoRaWAN Module

One of the key features of the LA66 module is its long-range capability[19]. It can transmit data over distances of up to several kilometers, allowing devices to communicate even in remote areas. This makes it an excellent choice for applications that require extended coverage and reliable connectivity.

Another notable feature of the LA66 module is its low-power consumption. It operates on low-energy protocols, which significantly reduces power consumption and extends the battery life of connected devices. This makes it suitable for applications that require long-term operation without frequent battery replacements.

The LA66 module also supports the LoRaWAN protocol, which is an open standard for low-power, wide-area networks. This enables seamless integration with existing LoRaWAN infrastructure and ensures interoperability with other LoRaWAN devices. It also provides secure and reliable communication through advanced encryption algorithms.

Additionally, the LA66 module offers support for multiple communication interfaces, such as UART, SPI, and I2C, providing versatility in its integration with various microcontrollers and sensors.

In summary, the LA66 LoRaWAN module is a versatile and efficient solution for IoT applications that require long-range connectivity, low-power operation, and secure communication.

4.1.4 LoRaWAN Gateway

A LoRaWAN gateway is a crucial component in a LoRaWAN network. It acts as a bridge between the end devices and the network server, enabling communication over long distances.

Figure 17 LoRaWAN Gateway

One of the primary functions of a LoRaWAN gateway is to receive data from multiple end devices through their respective LoRaWAN radios. It then forwards this data to the network server. This allows for efficient and reliable communication between the end devices and the server.

To ensure optimal performance, a LoRaWAN gateway requires a reliable power source and a stable internet connection. It is also important to consider the gateway's coverage range, as this determines the maximum distance over which end devices can communicate with it.

Furthermore, LoRaWAN gateways often come equipped with advanced features such as encryption and authentication mechanisms to ensure secure communication. This is particularly important when transmitting sensitive data or operating in environments where privacy is paramount.

In terms of deployment, LoRaWAN gateways can be either indoor or outdoor. Indoor gateways are typically smaller and more compact, making them suitable for deployment within buildings or enclosed spaces. Outdoor gateways, on the other hand, are designed to withstand harsh weather conditions and can cover larger areas.

In conclusion, a LoRaWAN gateway plays a vital role in facilitating communication within a LoRaWAN network. It acts as a bridge between end devices and the network server, allowing for efficient data transfer over long distances. With support for multiple channels, encryption mechanisms, and options for indoor or outdoor deployment, LoRaWAN gateways offer flexibility and reliability in IoT applications.

4.1.5 Power Bank

A power bank is a portable device that can provide backup power to charge electronic devices such as smartphones, tablets, and laptops. It is essentially a battery pack that can store electrical energy and release it when needed.

Figure 18 Power Bank (Storage Device)

Power banks come in various capacities, typically measured in milliampere-hours (mAh) or watt-hours (Wh). The capacity determines the amount of energy the power bank can store, which directly correlates to the number of charges it can provide to a device.

Power banks have become popular accessories due to their convenience and ability to provide backup power for electronic devices. They offer a portable and reliable solution for staying connected and powered up when access to a power source is limited, making them an essential companion for individuals on the go.

4.1.6 Jumper Wires

Jumper wires are an essential component in electronic circuits and prototyping. They are used to connect components on a breadboard or PCB, allowing for the flow of electricity between them. Jumper wires are typically made of flexible insulated wire with metal pins at each end, which can be easily inserted into the holes of a breadboard or soldered onto a PCB.

Figure 19 Jumper Wires

There are various types of jumper wires available, including male-to-male, female-to-female, and male-to-female. Male-to-male jumper wires have pins on both ends and are commonly used to connect two male headers or pins together. Female-to-female jumper wires have sockets on both ends and are used for connecting two female headers or pins. Male-to-female jumper wires have a pin on one end and a socket on the other, making them suitable for connecting male headers to female headers.

In conclusion, jumper wires play a vital role in electronic circuits and prototyping by allowing for the connection of components. Understanding the different types, lengths, colors, and gauges of jumper wires is crucial for creating efficient and organized circuits.

4.1.7 Micro-USB Cable

A micro-USB cable is a small and versatile device that has become a standard for charging and data transfer for many electronic devices. It is designed to connect devices such as smartphones, tablets, and Nucleo Microcontroller to computers or power sources, providing a reliable and efficient way to transfer data and charge devices.

Figure 20 Micro-USB Cable

In terms of functionality, the micro-USB cable supports fast charging and high-speed data transfer. This means that users can quickly charge their devices and transfer large files without having to wait for long periods of time.

Overall, the micro-USB cable is a reliable and efficient tool for charging and data transfer.

4.1.8 RN2483 Module

The RN2483 module is a highly versatile and efficient device that is used for long-range, low-power communications[20]. It is particularly well-suited for applications in the Internet of Things (IoT) industry. The module operates in the 868 MHz or 915 MHz frequency band, allowing for long-range communication capabilities. It utilizes the LoRaWAN protocol, which enables secure and efficient data transmission.

Figure 21 RN2483 Temp and Humidity Sensor Module.

RN2483 module is its low power consumption. It is designed to operate on very low power, making it ideal for battery-powered IoT devices. This allows for extended battery life and reduces the need for frequent battery replacements.

The module also boasts an impressive range, capable of reaching up to several kilometers in open spaces. This makes it suitable for applications that require long-distance communication, such as agricultural monitoring or smart city infrastructure.

Security is a top priority when it comes to IoT devices, and the RN2483 module takes this into consideration. It incorporates encryption algorithms to ensure secure data transmission, protecting against potential cyber threats.

In summary, the RN2483 module is a versatile and efficient device that offers long-range, low-power communication capabilities. With its ease of integration and security features, it is an ideal choice for IoT applications. Whether it's agricultural monitoring or smart city infrastructure, the RN2483 module provides a reliable solution for long-distance communication needs.

4.1.9 LHT65 Module

The LHT65 module, developed by Dragino Technology, is a versatile LoRaWAN-based wireless sensor module designed for environmental monitoring applications. It includes a variety of sensors for measuring critical environmental parameters such as temperature, humidity, and atmospheric pressure[21]. This makes it an excellent choice for weather monitoring, agriculture, and indoor climate control applications.

Figure 22 Dragino LHT 65 Sensor

One of the standout features of the LHT65 module is its exceptional battery life. It incorporates advanced power-saving technologies, ensuring long-lasting performance even in low-power applications. This makes it an ideal choice for remote monitoring or tracking systems where extended battery life is crucial.

Another notable feature of the LHT65 module is its user-friendly interface and easy integration capabilities. It comes with a comprehensive software development kit (SDK) that simplifies the integration process and allows for seamless communication with different platforms such as ThingsNetwork.

4.1.10 MCP1700 Voltage Regulator

The MCP1700 is a highly efficient and versatile voltage regulator manufactured by Microchip Technology[14]. This low dropout voltage regulator (LDO) is designed to maintain a stable and regulated output voltage, even when the input voltage is very close to the desired output voltage.

This voltage regulator is particularly well-suited for applications where power efficiency is crucial, such as battery-powered devices, portable electronics, and low-power systems. The MCP1700 also includes essential protection features like current limiting and thermal shutdown to ensure safe operation.

Figure 23 MCP 1700 Voltage Regulator

It's important to note that specific specifications and features may vary depending on the exact part number and variant of the MCP1700 voltage regulator, so it's essential to refer to the datasheet and documentation for the specific part you are using for detailed information and application guidelines.

4.1.11 Capacitor

The capacitor is commonly used in electronic circuits for various purposes, including voltage smoothing, noise filtering, timing applications, and energy storage. It is typically an electrolytic capacitor with a polarity, meaning it has a positive and negative terminal that must be connected correctly to prevent damage.

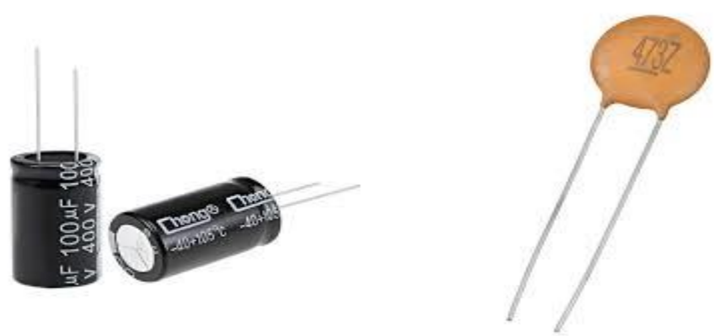

Figure 24 Different types of Capacitors.

The 100 uF capacitor finds use in a wide range of electronic devices and applications, playing a crucial role in stabilizing power supplies, filtering out unwanted signals, and managing energy storage needs.

4.1.12 USB-to-Serial Cable

A USB TTL Converter Cable, often referred to as a USB to TTL Serial Cable or simply a TTL cable, is a versatile hardware tool used for establishing communication between a computer or USB-equipped device and other electronic devices or microcontrollers that use TTL-level serial communication. These cables serve as essential bridges, enabling data exchange between computing devices and embedded systems, sensors, or microcontroller-based projects.

Figure 25 USB to Serial UART 5V TTL Header Cable

The FTDI (Future Technology Devices International) Chip TTL-232R USB (Universal Serial Bus) to Serial UART Converter Cables are versatile tools for connecting microcontrollers to USB-enabled devices. These cables feature the FT232R chip and a USB-A connector, making them easy to use and compatible with various systems. They draw power from the USB connection's +5V supply pins and offer Vcc connections at +5V DC. Users have the flexibility to select from available versions of the cable, which come with TTL logic levels of either +3.3V or +5V, based on their individual project requirements. For this particular project, a TTL logic level of +3.3V has been used.

4.2 Software

The following softwares are used in this project.

4.2.1 ThingsNetwork

The ThingsNetwork is a global, community-driven network that enables devices to connect to the internet using LoRaWAN (Low Power Wide Area Network) technology. It provides a decentralized infrastructure for IoT (Internet of Things) devices, allowing them to communicate with each other and with applications.

Figure 26 The ThingsNetwork Logo

One of the key features of The ThingsNetwork is its open-source nature, which encourages collaboration and innovation. Anyone can contribute to the network by setting up a gateway or building applications on top of it.

The network operates on a shared infrastructure model, meaning that gateways are owned and maintained by individuals or organizations who are part of the community. The ThingsNetwork also offers a range of services and tools to help developers build and deploy IoT applications. These include a cloud-based platform for managing devices and data, as well as an application marketplace where users can discover and share applications.

The ThingsNetwork has been used in a wide range of applications, from smart agriculture and environmental monitoring to asset tracking and logistics. Its low power consumption and long-range capabilities make it ideal for use cases where devices need to operate for extended periods without being connected to a power source.

Overall, The ThingsNetwork provides a powerful and flexible infrastructure for connecting IoT devices to the internet.

4.2.2 Setting up The Things Network

For transmitting the LoRaWAN Node data (sensor data) to TTN(The Things Network), it is necessary to learn the use of TTN in advance, such as how to set an application and the gateway. In this project, an indoor gateway that's compatible with LoRaWAN technology is used.

4.2.3 Register a Gateway

After configuring the gateway, it must be registered on the TTN website. Enter the TTN website: **https://console.thethingsnetwork.org/,** and log in your account. Then go to the console, select a cluster to start adding devices. In my case I selected EU cluster. In the Console, click on "Gateways" and then "Register Gateway. This involves adding gateway's unique identifier, known as its EUI or hardware address. TTN will use this information to associate your gateway with your account and enable it to connect to their network.

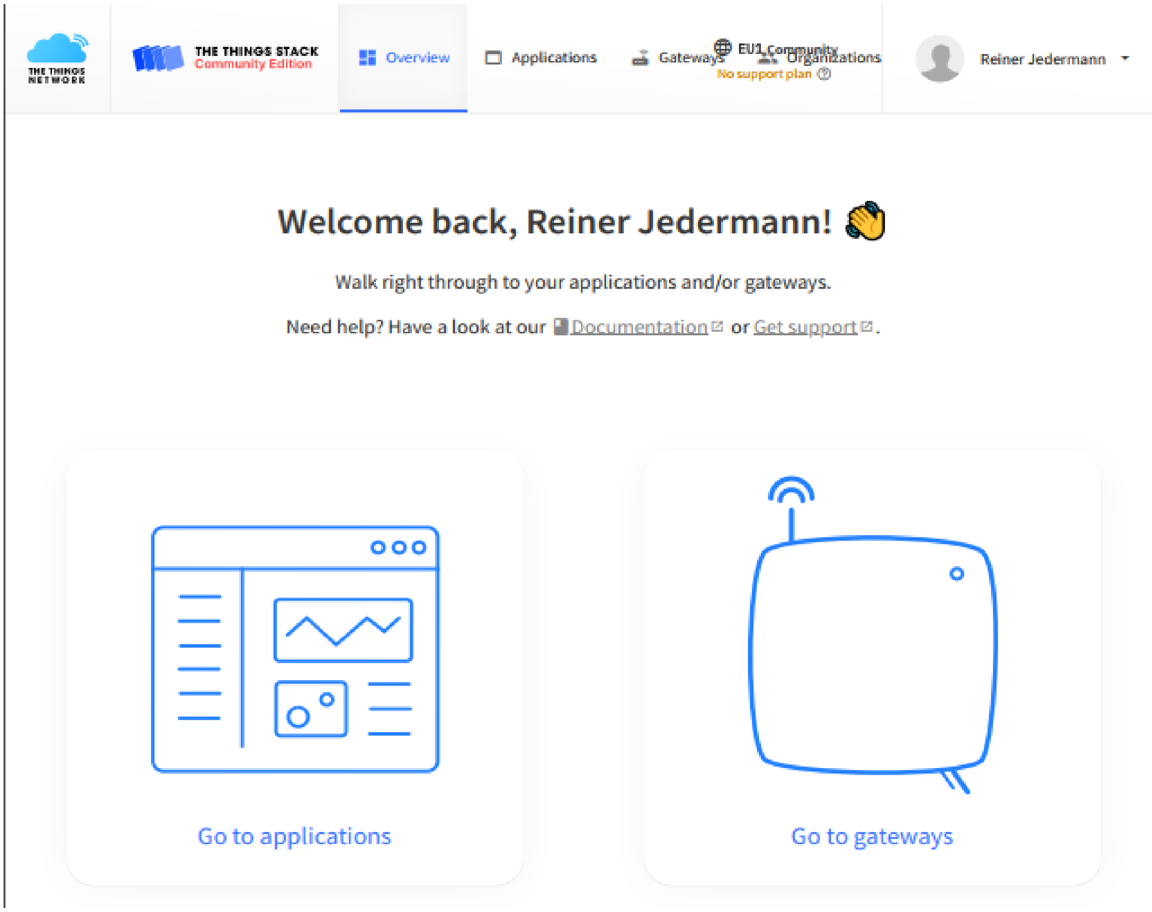

Figure 27 ThingsNetwork Console Page

4.2.4 Register Devices

Add a new application by selecting Application on the page. Then enter the application ID and additional details, and finally, create it.

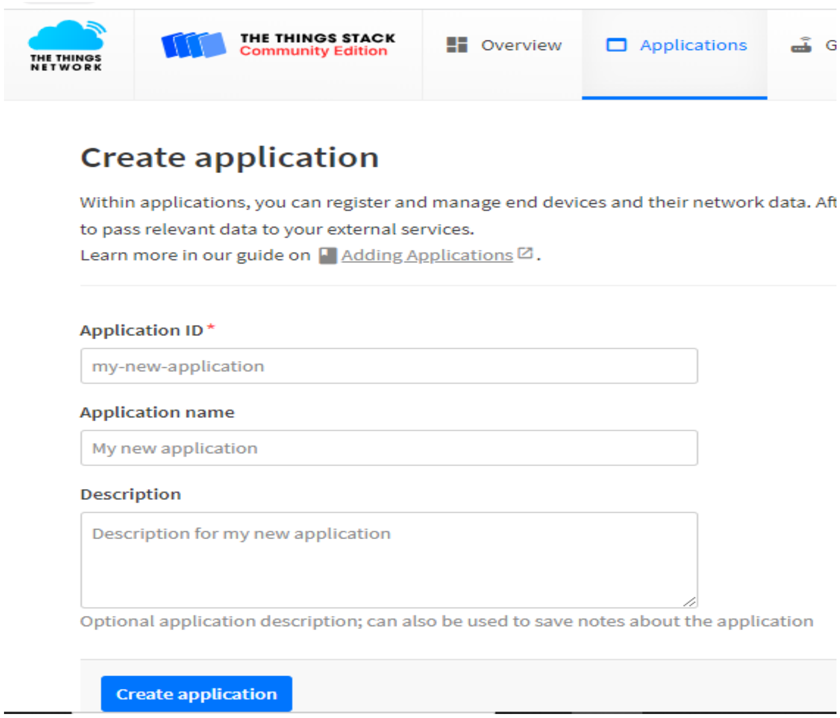

Figure 28 ThingsNetwork Create application Page

After the application is created, you can add devices to it. In the application overview, click on "Devices" and then "Register device." Enter a device ID and choose frequency plan, and the LoRaWAN version.

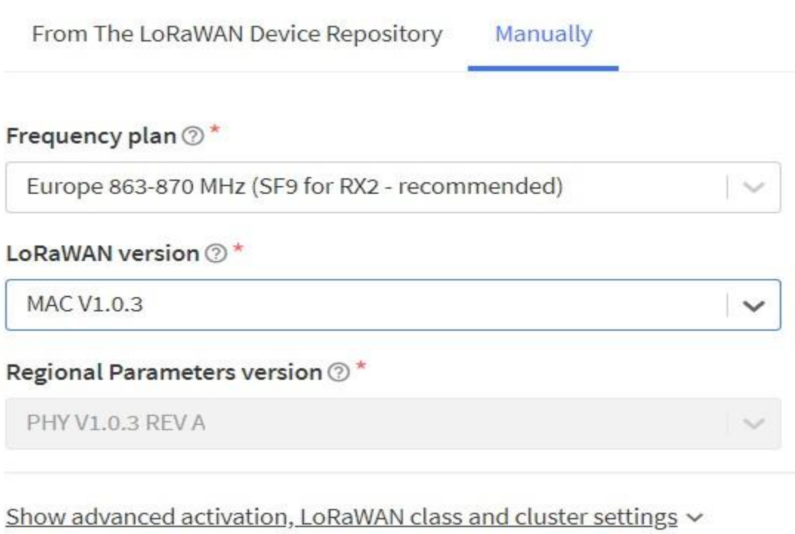

Figure 29 ThingsNetwork Device Frequency Plane.

Select manually to choose the active mode for OTAA and LoRaWAN version, then follow the reminders to type in the parameters. Then generate Device EUI, App EUI and AppKey. Then click on register end devices.

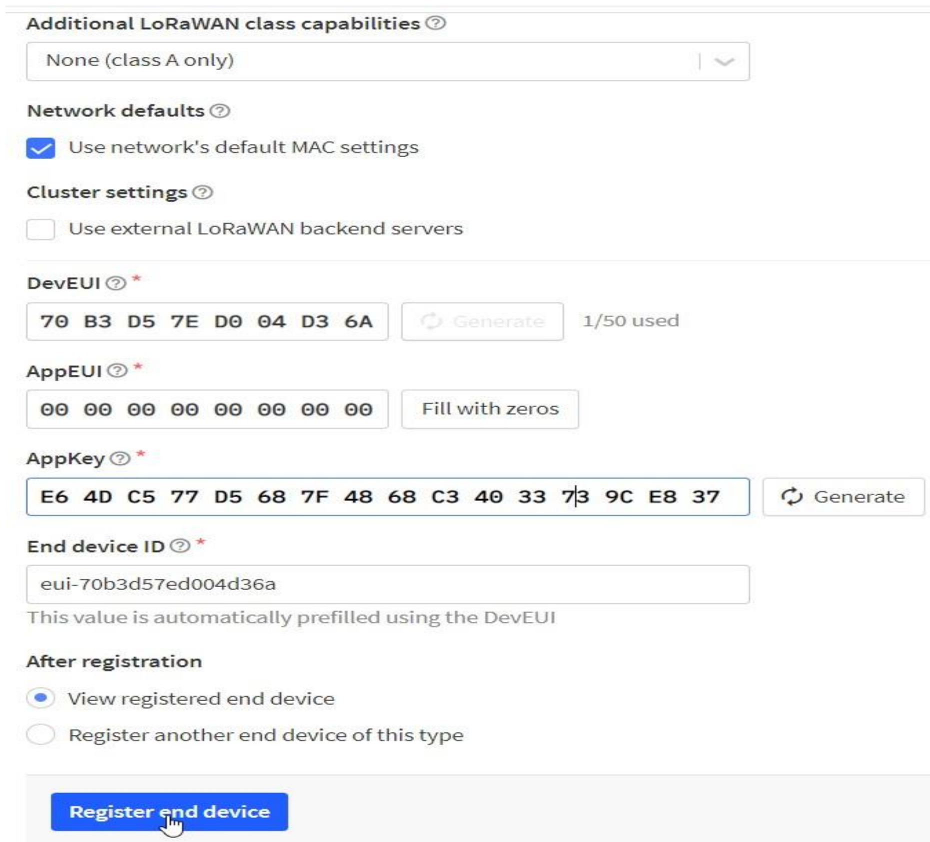

Figure 30 Device Register on ThingsNetwrok

Now the end node is successfully created. Hence you can see three parameters that had to be put into the code.

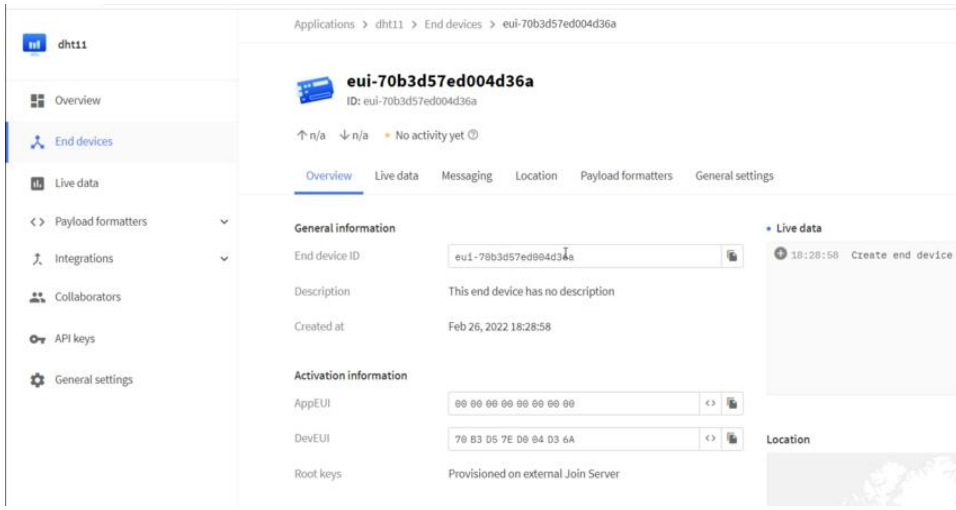

Figure 31 Register Devices on ThingsNetwrok

On the device page, it is necessary to set the payload formatters to decode the payload LoRaWAN node transmitted.

4.2.5 Node-RED

Node-RED is a versatile programming tool designed to facilitate the seamless integration of hardware devices, APIs, and online services[22]. With its user-friendly, browser-based editor, Node-RED simplifies the process of connecting these components by allowing users to create interconnected workflows using a diverse array of nodes available in its palette.

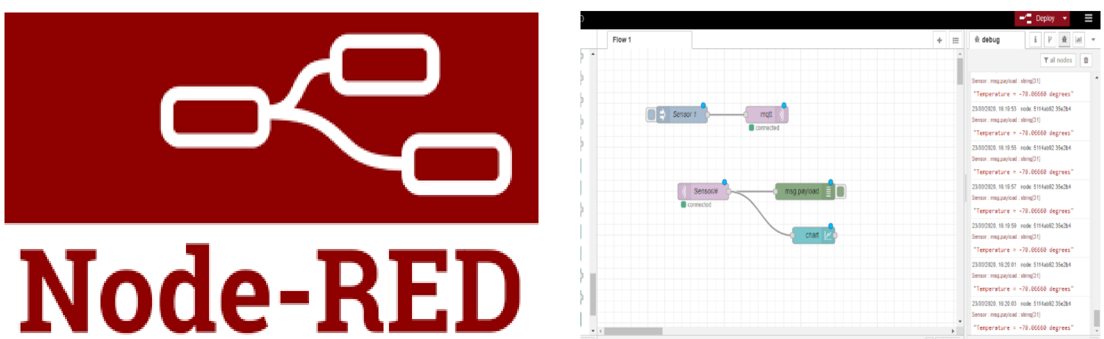

Figure 32 Node-RED Logo and Node-RED palette.

In this project, Node-RED has been employed to establish a connection between incoming data from The Things Network (TTN) and InfluxDB, an open-source database that excels in managing time-series data. This implementation enables the smooth and efficient transfer of data from TTN to InfluxDB, facilitating the storage and analysis of time-based information.

4.2.6 STM32 programming

In Nucleo Microcontroller, the most common language used is C, although some developers also use C++ or even assembly language for specific tasks. Having a good understanding of the chosen programming language is essential in order to write efficient and reliable code.

STM32 programming also involves using an integrated development environment (IDE) to write, compile, and debug the code. There are several IDEs available, such as STM32CubeIDE and Keil MDK. The STM32CubeIDE (Version: 1.13.1) was used in this project. This IDEs provide a range of tools and features to aid in the development process, including code editors, compilers, debuggers, and simulation environments. In addition to the IDE, developers often use software libraries provided by STMicroelectronics, the manufacturer of STM32 microcontrollers. These libraries contain pre-written code that can be used to simplify common tasks and accelerate development.

4.2.7 UART protocol

The UART (Universal Asynchronous Receiver/Transmitter) protocol is used in this project. It allows for serial communication between devices, enabling the transmission of data in a simple and efficient manner.

One of the key features of the UART protocol is its asynchronous nature. Unlike synchronous protocols, such as SPI or I2C, UART does not require a clock signal to synchronize data transmission[23]. Instead, it uses a start bit, followed by a fixed number of data bits and optional parity and stop bits to ensure accurate data transfer.

A transmitter and a receiver are typically used in UART communication. Additionally, UART supports a wide range of baud rates, allowing for flexible data transmission rates. Baud rate refers to the number of symbols transmitted per second and can be adjusted to accommodate different requirements. The baud rate used in this project was 9600 symbols per second to communicate between the Nucleo microcontroller and the LA66 LoRaWAN.

4.2.8 SPI (Serial Peripheral Interface) protocol

The SPI (Serial Peripheral Interface) protocol is a widely used communication standard in the field of electronics[23]. It allows multiple devices to communicate with each other using a synchronous serial interface. The SPI protocol consists of four main signals: MOSI (Master Out Slave In), MISO (Master In Slave Out), SCK (Serial Clock), and SS (Slave Select).

One of the key features of the SPI protocol is its ability to support full-duplex communication, meaning that data can be transmitted and received simultaneously. This makes it ideal for applications that require high-speed data transfer, such as memory chips, sensors, and display modules.

The SPI protocol operates in a master-slave configuration, where one device acts as the master and initiates the communication, while the other devices act as slaves and respond to the master's commands. The master device controls the clock signal and selects the slave device with which it wants to communicate using the SS signal. It supports different data formats, such as 8-bit, 16-bit, and 32-bit, allowing devices with different word lengths to communicate seamlessly.

To ensure reliable communication, the SPI protocol uses a simple protocol that includes a start bit, data bits, and an optional parity bit for error detection. The SPI protocol also supports various modes of operation, such as clock polarity and phase settings, which can be configured to meet the specific requirements of different devices.

Overall, the SPI protocol is a versatile and efficient communication standard that allows devices to exchange data quickly and reliably.

4.2.9 Subscribing messages using Python

Subscribing to MQTT messages using Python is a straightforward process. To get started, some Python libraries have to be downloaded. Paho-mqtt is one of the significant libraries which provides a simple API for interacting with MQTT brokers. This library can install using pip by running the command "pip install paho-mqtt".

After library installed, need to define a callback function that will be called whenever a new message is received. The payload is extracted from the received message and converted from base64 to hex format. The Pandas library can be used to display data in an eye-catching way.

4.2.10 InfluxDB

InfluxDB is a groundbreaking database management system that offers unique and powerful features for data storage and retrieval. Unlike traditional relational databases, InfluxDB is designed to handle large-scale datasets with high velocity and variety.

One of the key features of InfluxDB is its ability to handle time-series data efficiently. Time-series data, such as sensor readings or stock prices, can be stored and queried in real-time, allowing for near-instantaneous analysis. This makes InfluxDB particularly well-suited for applications in the Internet of Things (IoT) and financial industries.

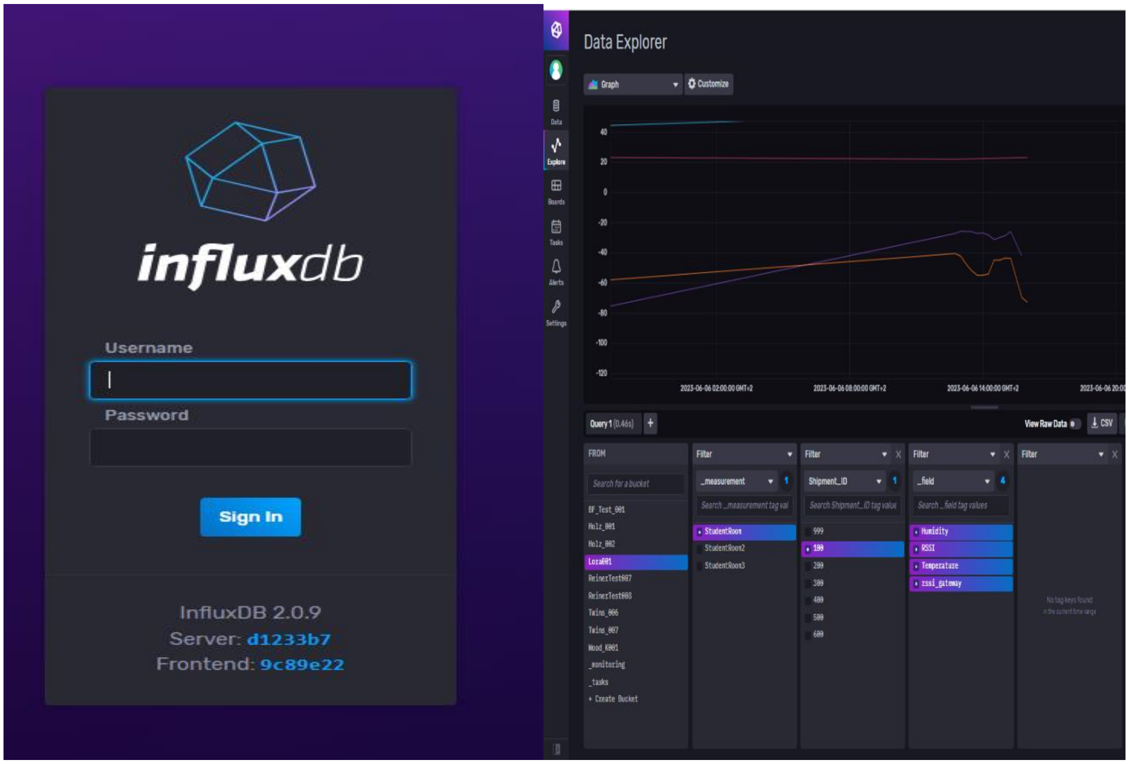

Figure 33 : InfluxDB

Figure 34 Sensors Value at InfluxDB

Real-time measurements of temperature, humidity, and rssi are sent to the influxDB database, which displays different visualizations and analyzes the data's pattern. InfluxDB also offers advanced analytics capabilities, including support for complex aggregations and machine learning algorithms. Users can perform powerful analytics directly within the database, eliminating the need to transfer large amounts of data to external systems. This not only saves time but also ensures data integrity and security.

Furthermore, InfluxDB provides a user-friendly interface and supports standard SQL queries, making it easy for developers and analysts to get started. It also offers integration with popular programming languages and frameworks, allowing for seamless integration with existing workflows.

5 Result Analysis

The primary objective has been to analyze the RSSI (Received Signal Strength Indicator) data during both uplink and downlink transmissions. In this project, a comprehensive sensor network is established, comprising a total of 15 different sensors utilized for uplink transmission. These sensors are strategically placed to collect data from various points. Additionally, two specialized sensors are deployed to facilitate downlink transmission, enabling bidirectional communication for enhanced data exchange and control. In the LA66 LoRaWAN module, data is transmitted at approximately 10-second intervals for both uplink and downlink transmissions.This frequent transmission interval ensures real-time data collection and responsiveness within the system.

Furthermore, the experiment has been conducted multiple times, involving varying locations and sensor configurations. This approach adds robustness to the study by accounting for different environmental conditions and sensor interactions, resulting in a more comprehensive and reliable dataset for analysis and research purposes.

5.1 Data Transmission

The experiment took place on the third floor of the NW1 building, within an extended corridor enclosed by concrete walls on both sides. At regular 10-meter intervals from the reference point, RSSI measurements were collected. Given that all measurements were conducted on the same floor, there was no need to employ the commonly used Pythagorean formula for distance calculation.

To determine the valid RSSI at each testing location, a robust methodology involved averaging all the values acquired throughout the testing process. This approach ensures a more precise assessment of signal strength throughout the testing area, accounting for environmental variations and signal fluctuations.

5.2 Received Signal Power and RSSI values

If a direct path exists between the end device (sensor) and the gateway in an environment free of signal interference and attenuation, the received signal power Pr, is inversely proportional to the square of the distance d[24].

$$P_r \propto d^{-2} \tag{1}$$

However, Equation 1 expresses the ideal relationship between RSSI and the relative distance. In the real world, a variety of factors, such as reflection, refraction, diffraction, and wave scattering from nearby objects, affect the value of the received signal strength.

An important factor in these applications is signal attenuation. There are several factors that contribute to signal attenuation. One of the main factors is distance. As the signal travels further, it naturally loses strength due to various physical effects, such as resistance and scattering. Signal attenuation can also be influenced by external factors, such as interference from other electronic devices or environmental conditions[24]. Radio signal strength can be attenuated with distance and could be define by equation (2)

$$\text{Attenuation} = \frac{signal\ strength(dB)}{distance(m)} \tag{2}$$

Empirical evidence indicates that a wall can reduce signal power by about 3dBm on average [1]. The received power may decay at a faster rate due to multi-path fading and non-uniform radio signal propagation. This transfers the relationship between Pr and d to:

$$P_r \ \infty\ d^{-n} \tag{3}$$

Where, n denotes the loss exponent.

The relationship between RSSI and distance in wireless communication is complex due to factors such as signal attenuation, interference, and multipath propagation. To accurately determine the relationship between RSSI and distance, various techniques such as calibration and modeling are employed. These techniques involve collecting RSSI data at known distances and using statistical analysis to derive mathematical models that approximate the relationship. The radio distance estimation model proposed by Texas Instruments is as follows [1]:

$$\text{RSSI} = -(10 \times \text{n}) \log10(\text{d}) - \text{A} \tag{4}$$

In Equation 4, RSSI is the radio signal strength indicator in dBm, n is the signal propagation constant or exponent is the relative distance between the communicating nodes which value is 3-5 for building[25] , and A is a reference received signal strength in dBm.

5.3 Curve fitting with RSSI values

Curve fitting is a mathematical technique used to find a line or curve that best represents a set of data points. The main objective of curve fitting is to create a model that accurately represents the data and can be used to make predictions or analyze trends.

There are several methods of curve fitting, including linear regression, polynomial regression, and non-linear regression. Linear regression involves finding the best-fit line that represents the data points. Polynomial regression involves finding the best-fit curve that represents the data points. Non-linear regression is used when the relationship between the independent and dependent variables is not linear. In this project, the non-linear regression curve fitting is used. The least squares method is a popular approach in curve fitting. It is used to find the best-fitting curve that minimizes the sum of the squared residuals between the observed data points and the estimated curve. The least squire method is also used in this project. The following code was used to calculate the reference received signal strength (A) and signal propagation constant (n).

```python
from scipy.optimize import curve_fit

def f(x, A, n):
    return (-A-10*n*np.log10(x))

par, var = curve_fit(f, x, y)

A, n = par
print('A:',A)
print('n:', n)
```

Figure 35 Curve Fitting

5.4 Test Setup

In this project, multiple sensors are connected to the gateway. multiple sensors play a crucial role in gathering data and transmitting it to a central gateway. These sensors are equipped with various functionalities that allow them to collect information from the environment they are deployed in. However, the effectiveness of these sensors relies on their ability to maintain a strong connection to the gateway.

A well-executed test setup is essential for project success. It ensures accurate measurement of performance, identifies potential issues, and allows for necessary improvements.The test setup for this uplink transmission is shown below.

Figure 36 Test Setup for Uplink Transmission.

By continuously monitoring these values, it becomes possible to detect patterns or anomalies that may indicate potential malfunctions or problems with the sensors or gateway.

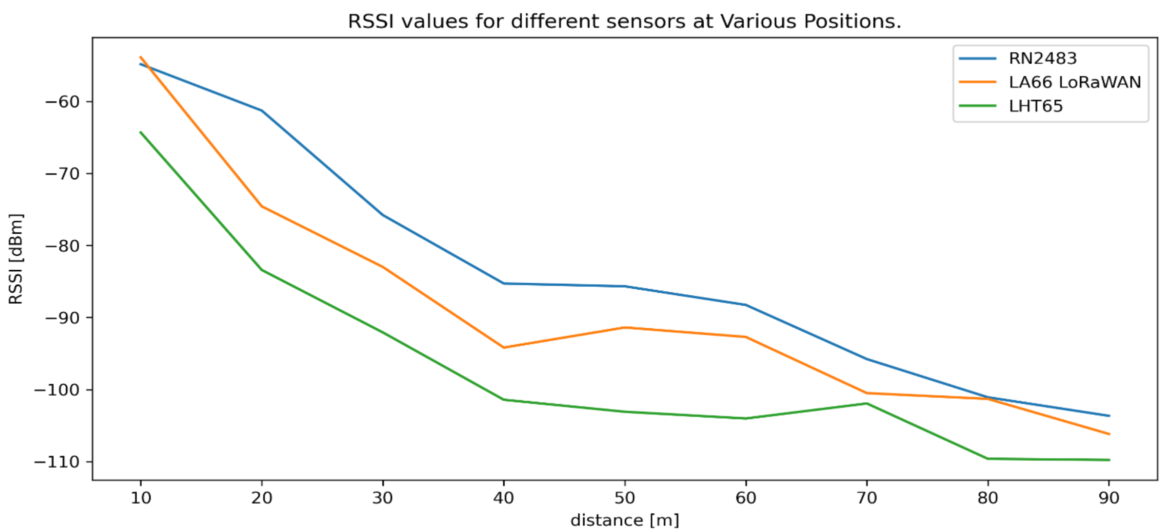

Figure 37 Change of RSSI values with Respect to Distance for Different Sensors.

The above figure shows a pattern in which the rssi values decrease as the distance increases for all sensors, which agrees with the theoretical study.

5.4.1 RSSI values for Uplink Transmission

In communication, the Received Signal Strength Indicator (RSSI) is a measurement of the power level of a received signal. It is commonly used to evaluate the quality and performance of uplink transmission in networks. It is typically represented as a numerical value in decibels (dBm) and is used to assess the quality of the signal reception. A higher RSSI value generally indicates a stronger signal and better connectivity, while a lower RSSI value suggests a weaker signal and potential issues with data transmission or reception.

However, it's important to note that RSSI values alone may not provide a complete picture of network performance. Factors such as interference, multipath fading, and environmental conditions can affect the accuracy of RSSI measurements.

5.4.2 RN2483 sensors

The RN2483 sensors are highly versatile and innovative devices that offer a range of features and capabilities. These sensors are specifically designed for the Internet of Things (IoT) applications and enable seamless connectivity and communication. One key feature of the RN2483 sensors is their low power consumption, which makes them ideal for battery-powered IoT devices.

5.4.3 Uplink RSSI values with RN2483 Modules

RSSI measurements were taken at intervals of 10 meters from the reference point. The table below represents the average rssi values for different RN2483 modules at different position.

	RN-1	RN-2	RN-3	RN-4	RN-5_PCB
10 meters	57.7	-52.2	-52.2	-54.1	-57.8
20 meters	-56.3	-56.8	-61.6	-64.3	-67.2
30 meters	-71.6	-82.7	-80.0	-71.8	-72.6
40 meters	-80.2	-86.9	-83.1	-85.1	-91.0
50 meters	-82.8	-86.0	-93.5	-82.4	-83.6
60 meters	-76.13	-89.8	-93.5	-94.4	-87.3
70 meters	-91.7	-92.5	-101.9	-97.4	-95.2
80 meters	-96.7	-102.6	-101.0	-98.1	-106.9
90 meters	-99.2	-107.4	-108.3	-99.6	-103.7

Table 2 Average RSSI values for RN2483 Module in Different Distance.

The bar graph provides a visual representation of the RSSI values, allowing users to easily compare and analyze the strength of different signals. The bar graph below displays the average RSSI of the RN2483 modules.

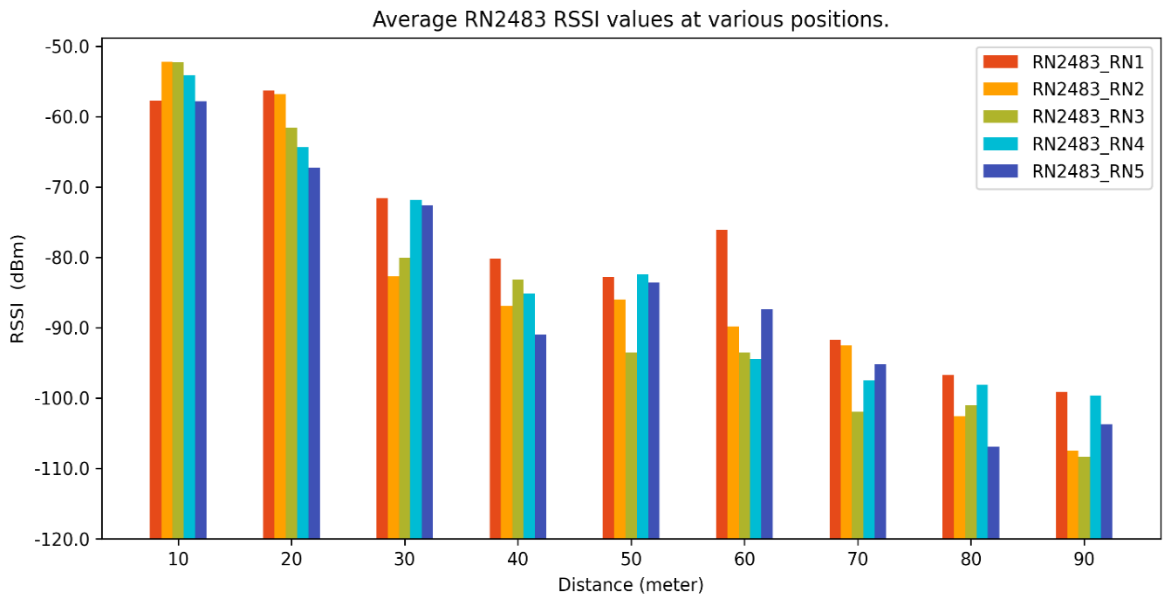

Figure 38 Strength of RSSI values for RN2483(Mean) Module in Different Distance.

The x-axis of the graph represents the signals being measured at different positions, while the y-axis represents the RSSI values. Each bar on the graph corresponds to a specific signal, and its height indicates the RSSI value. The higher the bar, the stronger the signal.

Based on the above graph, it is evident that the signals were stronger when devices were closer to the gateway and weakened as the distance from the gateway increased.

5.4.4 RN2483 Module with Curve Fitting

The RN2483 is a highly advanced module that is used for Low Power Wide Area Network (LPWAN) communication. It is designed to provide long-range communication while consuming minimal power. The module uses the LoRaWAN protocol. The formula in equation 4 can be used to calculate RSSI values.

By employing the linear curve fitting technique on the data from the RN2483 modules, the determined parameters are as follows: the reference received signal strength (A) is -3 dBm, and the signal propagation constant (n) is 5.32.

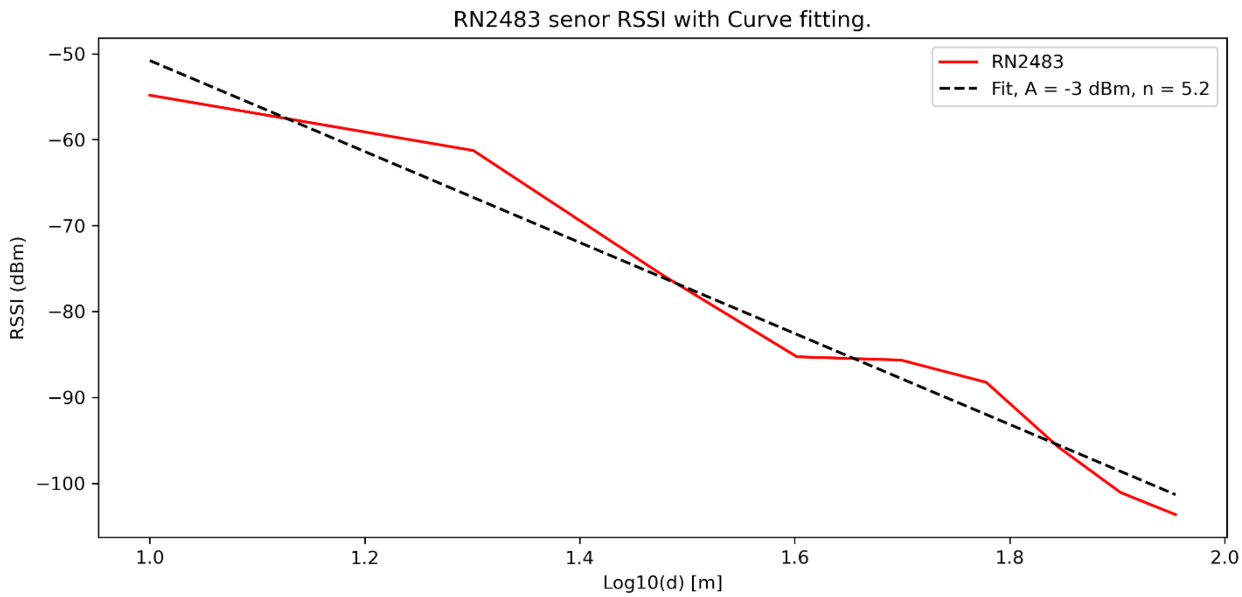

Figure 39 RSSI Line with Curve Fitting Line in RN2483 Module.

5.4.5 LA66 LoRaWAN Module

The LA66 LoRaWAN Module is a highly efficient and versatile device that is used for long-range communication. It supports the LoRaWAN protocol, which enables it to transmit data over long distances while consuming minimal power. The module is equipped with a high-performance RF transceiver, which allows for reliable and secure communication.

5.4.6 Uplink RSSI values with LA66 LoRaWAN Module

The LA66 LoRaWAN Module provides detailed information about RSSI values, such as the current RSSI value, signal-to-noise ratio (SNR), and link quality indicator (LQI). The table below illustrates the average RSSI values corresponding to various positions for distinct LA66 LoRaWAN modules.

	LA66_USB - 1	LA66_ USB -2	LA66_USB -3	LA66 -66	LA66 -35	LA66 -45
10 meters	-64.9	-53.2	-56.2	-53.2	-52.2	-43.4
20 meters	-82.8	-72.8	-73.5	-69.4	-71.0	-77.6
30 meters	-100.8	-80.5	-72.7	-78.2	-85.8	-79.6
40 meters	-101.2	-95.93	-94.2	-90.7	-96.2	-86.7
50 meters	-98.8	-92.0	-85.2	-86.1	-100.2	-85.7
60 meters	-105.5	-95.8	-91.4	-93.3	-90.4	-79.6
70 meters	-104.7	-100.4	-98.9	-101.0	-103.7	-94.1
80 meters	-108.8	-101.2	-100.6	-98.5	-102.5	-96.0
90 meters	-111.8	-108.0	-106.5	-106.8	-103.3	-100.5

Table 3 Average RSSI values for LA66 LoRaWAN Module in Different Distance

The bar graph visually depicts the RSSI values, facilitating straightforward comparison and analysis of signal strengths. The following bar graph showcases the average RSSI for both the LoRaWAN LA66 and LoRaWAN USB modules.

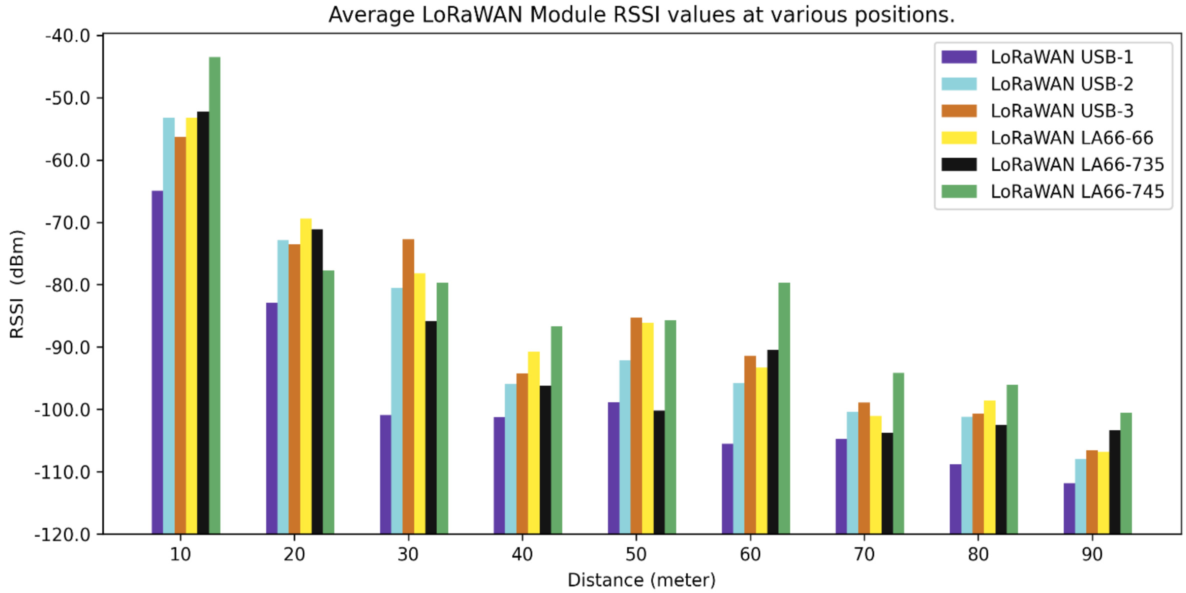

Figure 40 Strength of RSSI values for LA66 LoRaWAN(Mean) Module in Different Distance.

On the graph, the x-axis illustrates the positions where signals were measured, while the y-axis represents the RSSI values. Each bar on the graph corresponds to a particular signal, with its height denoting the RSSI value. A taller bar signifies a stronger signal.

From the presented graph, it's clear that signals exhibited greater strength when devices were in proximity to the gateway, gradually diminishing as the distance from the gateway increased.

5.4.7 LA66 LoRaWAN Module with Curve Fitting

One of the standout features of this LoRaWAN LA66 module is its curve fitting capability. This allows the module to accurately predict and optimize data transmission based on the changing conditions of the network. By analysing the received signal strength indicator (RSSI), it becomes possible to compute the unfamiliar variables of the reference received signal strength (A) and the signal propagation constant (n). The formula employed to compute the RSSI value is provided below:

$$\mathrm{RSSI} = -(10 \times n)\log 10(d) - A$$

Following the application of the linear curve fitting approach, the reference received signal strength (A) and signal propagation constant (n) for LoRaWAN have been determined as -6.3 dBm and 5.08, respectively.

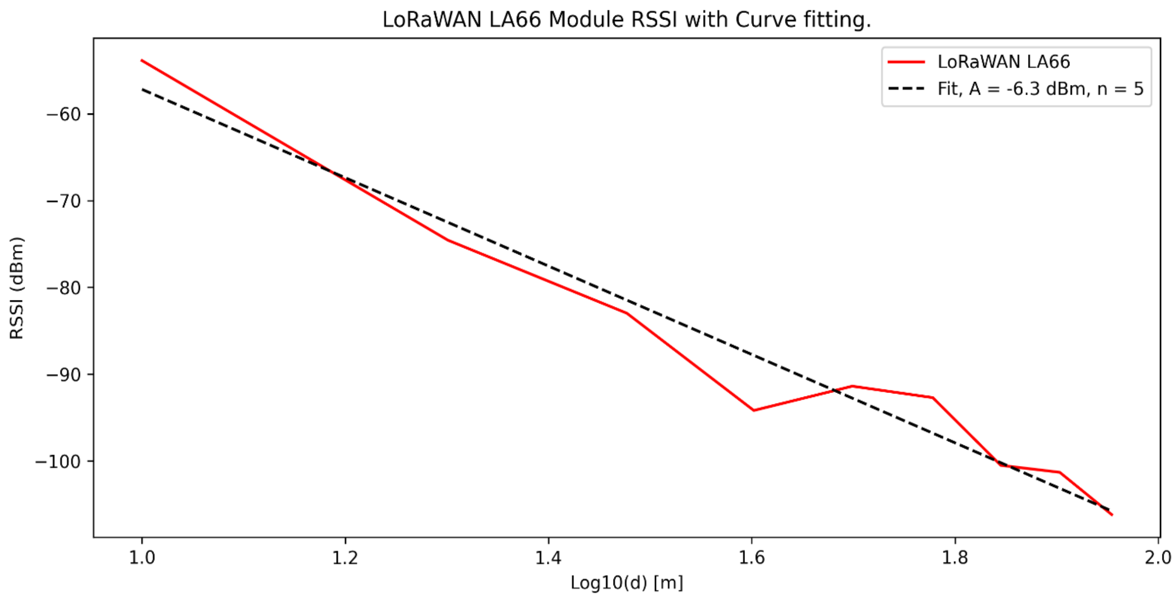

Figure 41 RSSI Line with Curve Fitting Line in LA66 LoRaWAN Module.

5.4.8 Distance covered by LA66 LoRaWAN Shield

The LA66 LoRaWAN Shield is capable of covering long distances with its advanced technology. It enables efficient communication between devices through the use of LoRaWAN protocol. With this shield, devices can transmit data over long ranges, even in areas with low signal strength. In the context of this project, real-time measurements were taken to determine the coverage range of the LA66 LoRaWAN Shield, revealing an impressive distance of approximately 1.5 km from the reference point at Riensberger Str. 108, 28359, Bremen. This capability to transmit data over such distances can be particularly advantageous for applications that require wide-ranging connectivity in challenging signal environments.

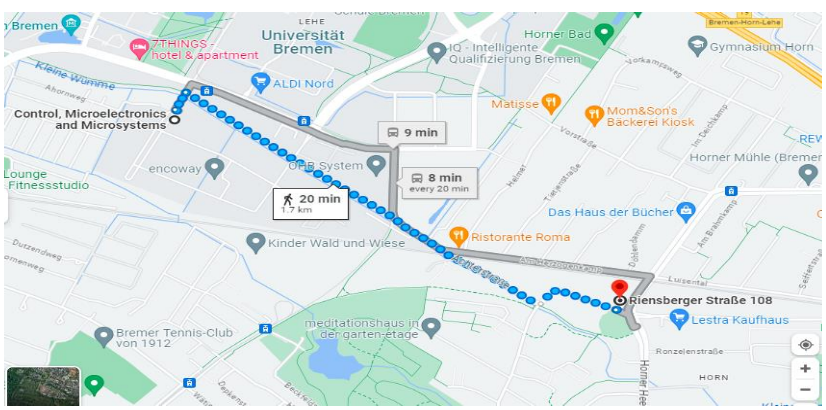

Figure 42 Map for Distance from Riensberger Str.108, 28359 to University of Bremen (NW1 Building)

5.4.9 Dragino LHT65

The LHT65 LoRaWAN sensor is a highly advanced device that offers a range of unique features and capabilities. With its compact design and long battery life, this sensor is ideal for a wide range of applications, including environmental monitoring, asset tracking, and smart agriculture.

5.4.10 Uplink RSSI values with Dragino LHT65 Modules

The RSSI values measured by the LHT65 LoRaWAN sensor can be used to determine the quality of the communication link between the sensor and the LoRaWAN gateway. The table below show the average RSSI values associated with different positions for individual Dragino LHT65 modules.

	LHT65-27A	**LHT65-86**	**LHT65-91**
10 meters	-63.1	-62.8	-66.9
20 meters	-83.8	-80.1	-86.2
30 meters	-86.6	-83.9	-105.6
40 meters	-99.7	-100.7	-103.8
50 meters	-97.8	-101.2	-110.2
60 meters	-97.9	-106.1	-108.0
70 meters	-96.4	-102.4	-106.9
80 meters	-109.2	-111.6	-108.0
90 meters	-109.6	-108.8	-110.8

Table 4 Average RSSI values for LHT65Module in Different Distance

The bar graph visually illustrates the RSSI values, enabling users to readily compare and analyse the signal strength variations. The following bar graph depicts the average RSSI of the Dragino LHT65 modules.

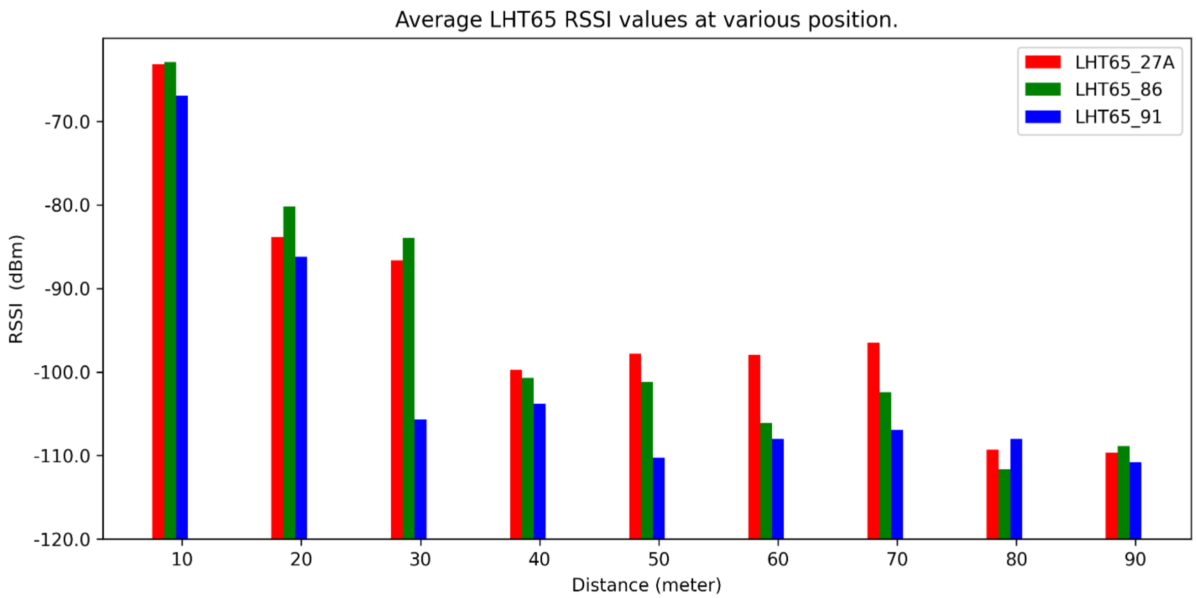

Figure 43 Strength of RSSI values for LHT65 (Mean) Module in Different Distance.

The graph's x-axis denotes the positions where signal measurements were taken, while the y-axis represents the corresponding RSSI values. Each bar on the graph corresponds to a specific signal, and the height of the bar indicates the RSSI value. A taller bar indicates a stronger signal.

From the graph provided, it's evident that signals displayed higher strength when devices were closer to the gateway, gradually weakening as the distance from the gateway increased.

5.4.11 Dragino LHT65 Module with Curve Fitting

By utilizing curve fitting technology in the Dragino LHT65 module, it becomes possible to calculate the unknown parameters of the reference received signal strength (A) and the signal propagation constant (n).

The formula provided below is utilized for calculating the RSSI values for Dragino LHT65 modules:

$$RSSI = -(10 \times n) \log10(d) - A$$

Utilizing curve fitting techniques, the calculated values for the reference received signal strength (A) and signal propagation constant (n) are 22.5 dBm and 4.6, respectively.

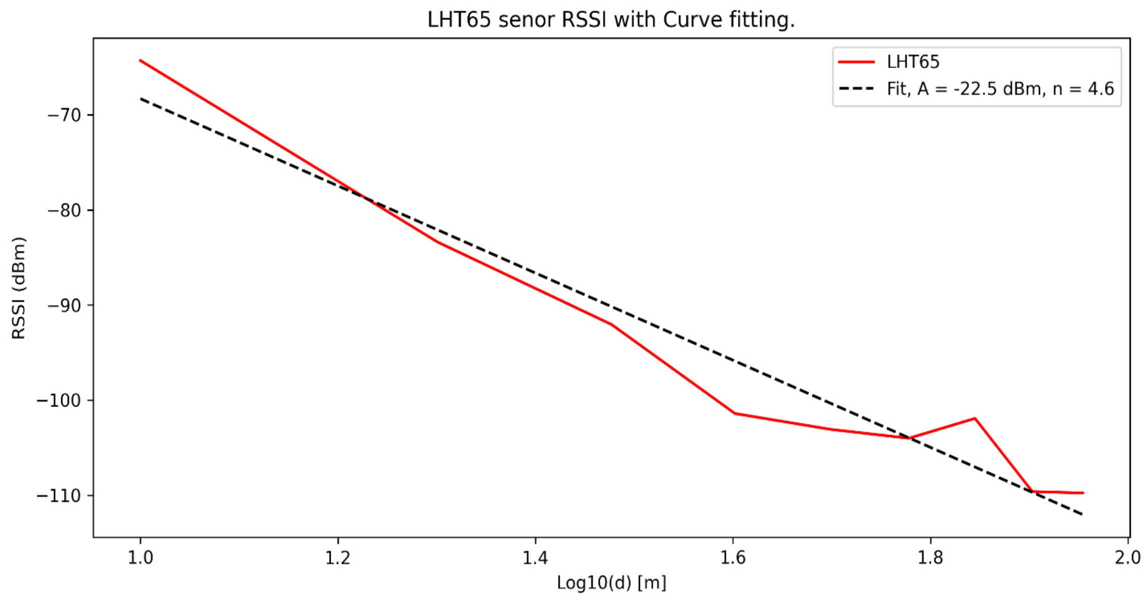

Figure 44 RSSI Line with Curve Fitting Line in LHT65 Module.

5.4.12 Reference Received signal Strength (A) At Constant Signal Propagation for LA66 LoRaWAN

The parameter 'n,' denoting the constant signal propagation, plays a pivotal role in evaluating RSSI values across varying Received Signal Strengths (A). This 'n' value serves as an indicator of the path loss exponent, providing a quantitative measure of the rate at which signal strength diminishes as the distance between transmitter and receiver increases. This exponent is critical in understanding how effectively signals propagate and attenuate through different environmental conditions, aiding in the optimization of communication systems and network performance.

	Received Signal Strength (A)	**Signal Propagation**
LA66 LoRaWAN_USB_2	-8 dBm	5
LA66 LoRaWAN_USB_3	-6 dBm	5
LA66 LoRaWAN_USB_72F	-17 dBm	5
LA66 LoRaWAN_735	-8dBm	5
LA66 LoRaWAN_745	-2 dBm	5

Table 5 Received Signal Strength in Constant Signal Propagation for LA66 LoRaWAN Module

Observing the table, it becomes evident that signal strength remains relatively stable for consistent signal propagation. Notably, the LA66 LoRaWAN_USB_72F module exhibits a more significant signal loss compared to the other LA66 LoRaWAN modules, resulting in a higher received signal strength(A) for this particular module.

5.4.13 Reference Received signal Strength (A) At Constant Signal Propagation for RN2483 Module

The following table presents the Reference Received Signal Strength (A) within a scenario of consistent signal propagation (n) for RN2483 Module.

	Received Signal Strength (A)	Signal Propagation
RN2483_1	1 dBm	5
RN2483_2	-2 dBm	5
RN2483_3	-3 dBm	5
RN2483_4	-2 dBm	5
RN2483_PCB	-4 dBm	5

Table 6 Received Signal Strength in Constant Signal Propagation for RN2483 Module

Upon reviewing the table, it's clear that signal strength remains relatively consistent under conditions of constant signal propagation n = 5.

5.4.14 Reference Received signal Strength (A) At Constant Signal Propagation for LHT65 Module

The table below illustrates the Reference Received Signal Strength (A) in a situation where signal propagation (n) remains constant, specifically for the LHT65 Module.

	Received Signal Strength (A)	Signal Propagation
LHT65_27A	-15 dBm	5
LHT65_86	-14 dBm	5
LHT65_91	-20 dBm	5

Table 7 Received Signal Strength in Constant Signal Propagation for LHT65 Module

Upon examining the table, it is apparent that the signal strength for the RN2483 Module remains high compared to other modules. This observation aligns with our findings in Section 9, where we observed a higher occurrence of packet loss in the LHT65 module which support our study.

5.4.15 Reference Received signal Strength (A) At Constant Signal Propagation:

The table below provides the average Reference Received Signal Strength (A) under the conditions of constant signal propagation (n).

	Received Signal Strength (A)	**Signal Propagation**
RN2483 Module	-2 dBm	5
LA66 LoRaWAN	-8 dBm	5
Dragino LHT65	-17 dBm	5

Table 8 Received Signal Strength in Constant Signal Propagation for Different Sensors.

The table presented above illustrates the variation in Received Signal Strength (RSS) across a consistent signal propagation value of 5. It reveals that RSS values fluctuate within the range of -17 dBm to -2 dBm, with a notable difference of approximately -15 dBm between them. This data provides valuable insights into the signal behaviours observed in different sensors.

The figures below illustrate the Received Signal Strength (RSS) with a consistent signal propagation value of 5.

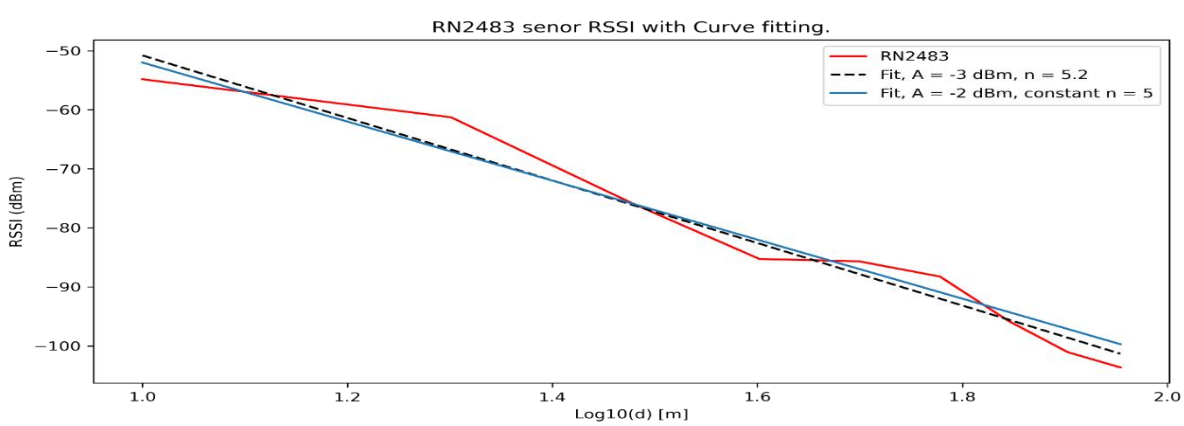

Figure 45 RSSI Line, Curve Fitting Line, and Curve Fitting line with Constant Signal Propagation in RN2483Module

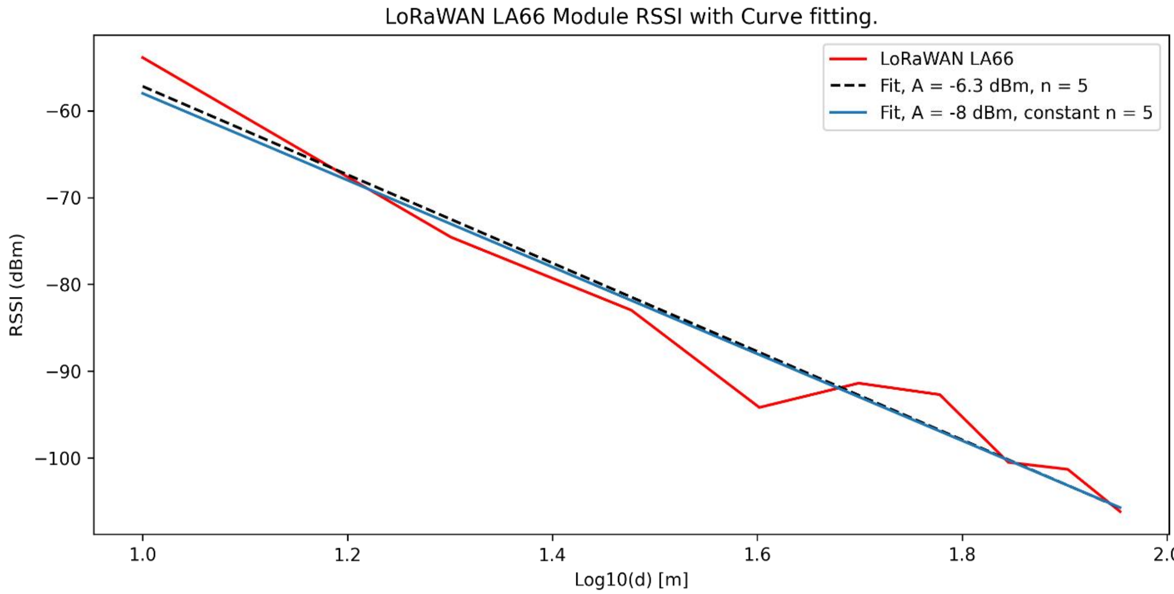

Figure 46 RSSI Line, Curve Fitting Line, and Curve Fitting line with Constant Signal Propagation in LA66 LoRaWAN Module

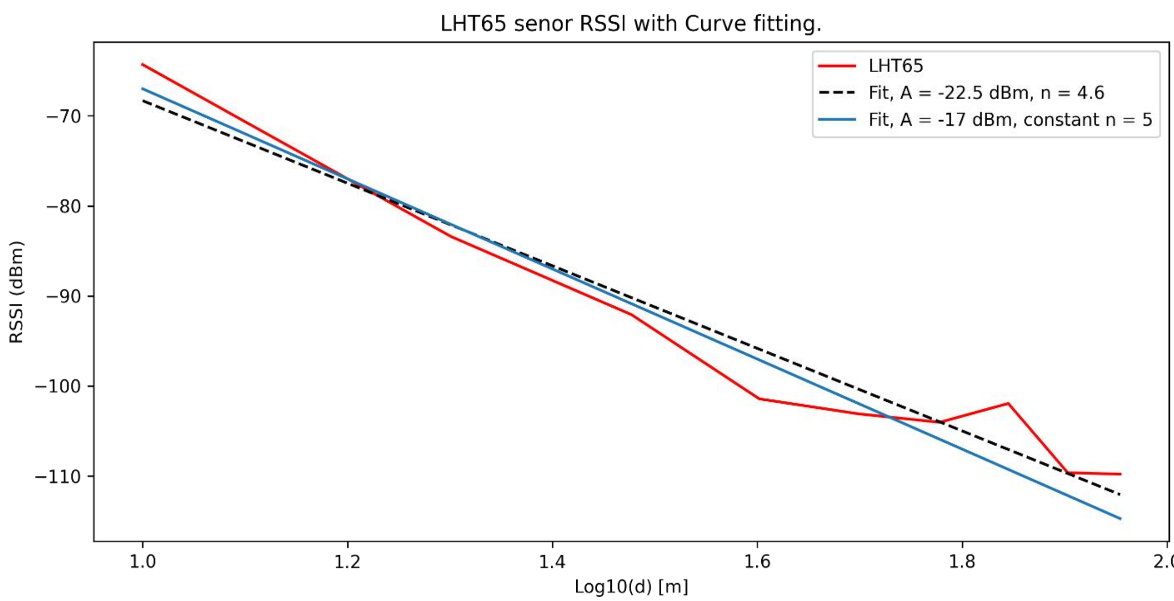

Figure 47 RSSI Line, Curve Fitting Line, and Curve Fitting line with Constant Signal Propagation in LHT65 Module

5.5 RSSI values for Downlink Transmission

The RSSI (Received Signal Strength Indicator) values play a crucial role in downlink transmission. These values are used to measure the strength of the signal received by a device from a base Gateway. In downlink transmission, the Gateway sends data to the device, and the RSSI values help determine the quality of the signal. The RSSI values are usually expressed in dBm (decibel-milliwatts) and can vary depending on several factors. One such factor is the distance between the device and the Gateway. As the distance increases, the signal strength decreases, resulting in lower RSSI values.

5.5.1 Test Setup for Downlink Transmission

In the context of downlink transmission, both LA66 LoRaWAN and RN2483 modules are employed. Downlink transmission involves the process of sending data from a central gateway to remote devices or sensors, enabling bidirectional communication and control in various IoT and remote monitoring applications.

Below is the test setup for downlink transmission.

Figure 48 Test Setup for Downlink Transmission.

5.5.2 LA66 LoRaWAN Module

The LA66 LoRaWAN module is a powerful device designed for efficient uplink and downlink transmission. It utilizes the LoRaWAN technology, which enables long-range and low-power communication. In addition to low power consumption, the LA66 module offers a long-range communication capability. It can transmit data over distances of several kilometers, making it suitable for applications that require wide coverage. The module operates in the 868 MHz frequency band, which is commonly used in Europe.

5.5.3 Downlink RSSI values with LA66 LoRaWAN Module

The LA66 LoRaWAN Module is capable of providing downlink RSSI values, which are essential for measuring the signal strength and quality of a LoRaWAN network. The table exhibits the RSSI values of the LA66 LoRaWAN sensor recorded at different positions. In total, 104 data points were collected, involving the acquisition of an average of 10 readings per position. This comprehensive dataset allows for a thorough analysis of signal strength across multiple locations, enhancing our understanding of signal behavior in various environmental conditions.

Sensor RSSI (dBm)	Gateway RSSI (dBm)	Distance (meter)
-45.3	-64.4	10
-52.4	-71.9	20
-63.0	-83.0	30
-70.8	-97.9	40
-80.8	-99.0	50
-85.7	-100.1	60
-90.5	-101.3	70
-87.7	-102.0	80
-113.3	-113.2	90

Table 9 Sensor and Gateway RSSI (Average) with Respect to Distance.

The distance is the most important factor in determining RSSI values. When the distance increased, some Gateway RSSI values were lost. The lost RSSI values were replaced with -99999.

Users can quickly compare and evaluate the strength of various signals by using the bar graph's visual representation of the RSSI values.

This bar graph also illustrates the change of RSSI strength with the distance.

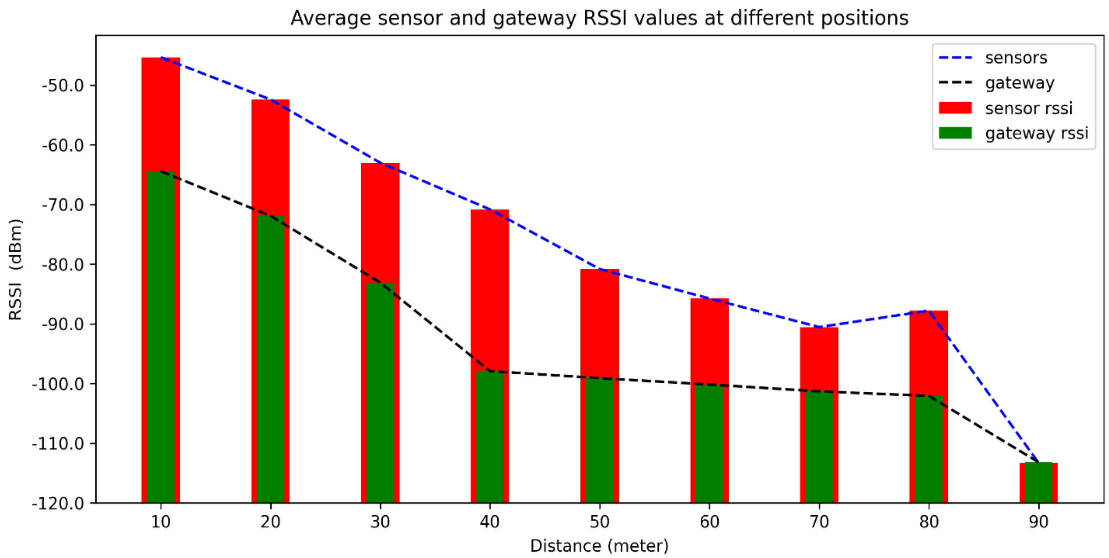

Figure 49 Sensor and Gateway RSSI (Mean) values at different positions.

The graph's x-axis represents the various signals being measured at various positions, while the y-axis represents the RSSI values. Each bar on the graph represents a different signal, and its height represents the RSSI value. The stronger the signal, the higher the bar.

According to the bar graph, the strength of the RSSI value is strong (small in value) near the Gateway and decreases as the distance increases. In a 50-meter radius, the average RSSI value is -75dBm where the minimum RSSI value is – 45 dBm at 10 meters and maximum value is -113 dBm at 90 meters.

5.5.4 Downlink Sensor RSSI values with Moving Average

A moving average is a statistical calculation that is used to analyze data and identify trends over time. A moving average is computed by averaging a set of data points within a specified window or interval. The window size used in this calculation was 10. The moving average was plotted on top of the RSSI graph, allowing us to observe trends and smooth out data fluctuations. It was discovered from the moving average graph that the graph has a decreasing trend, which means that as the distance increased, the RSSI values decrease as well.

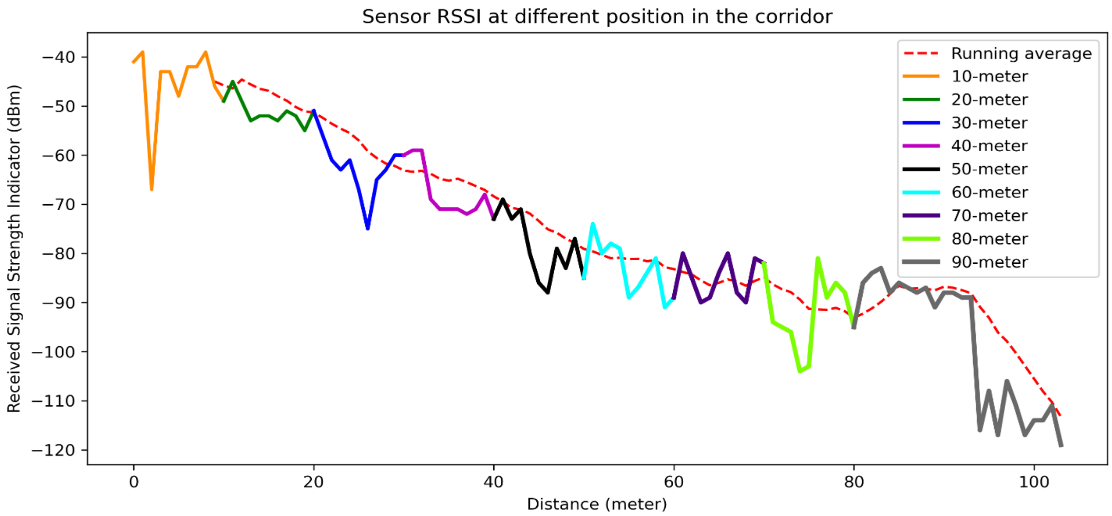

Figure 50 Sensor RSSI values with Moving Average.

1.1.1 Relationship between Downlink Sensor RSSI and Gateway RSSI

The RSSI values for the sensor downlink signal and the Gateway signal have a strong correlation. The correlation was approximately 90%.

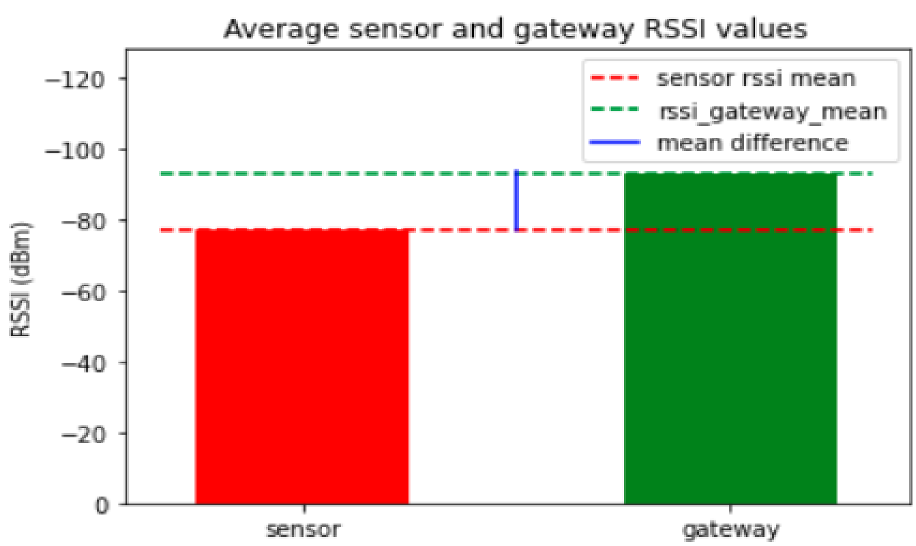

Figure 51 Difference Between Average sensor and gateway RSSI values.

From the bar graph it was seen that, the downlink received sensor signal rssi values are stronger than the gateway rssi values, as shown in the above figure. The RSSI difference between the received downlink signal and the Gateway signal was approximately -16 dBm. The mean or average of the received sensor signal on the downlink at different positions was -76.9 dBm. Similarly, the RSSI of the received gateway signal was -92.4 dBm on average.

Additionally, there is a negative correlation between the RSSI of the received downlink signal and distance, which means that higher RSSI values correspond to shorter distances and lower RSSI values correspond to longer distances. This relationship, however, can be influenced by a variety of factors such as obstacles, interference, and environmental conditions. The difference between them also became smaller as the distance grew, and after 90 meters from the reference point, they began to overlap.

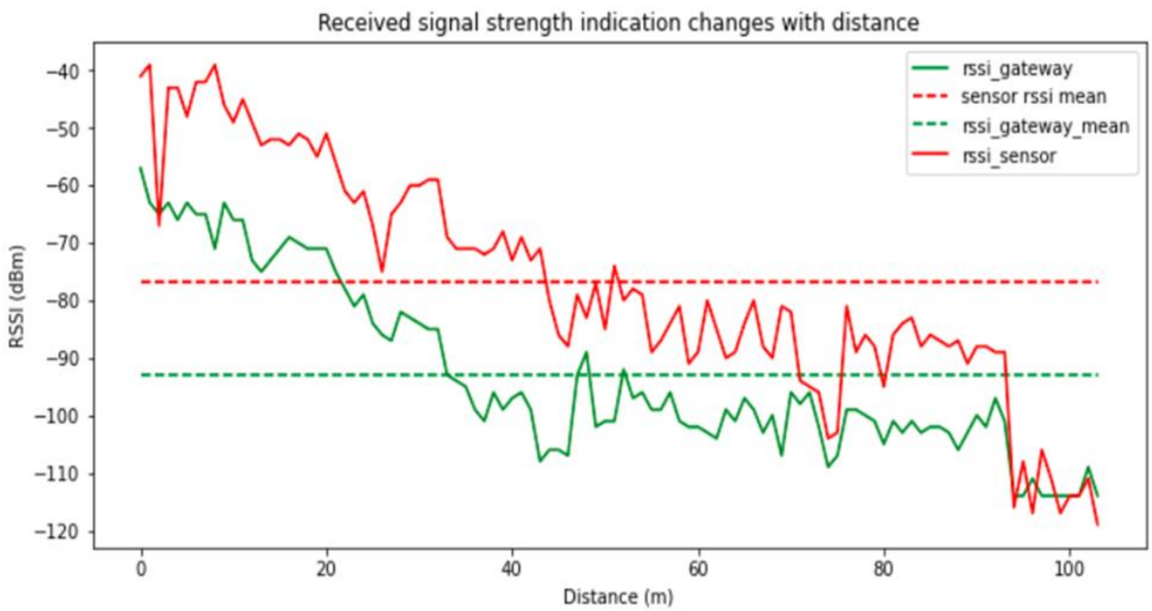

Figure 52 Change of Sensor and Gateway RSSI with distance with their Mean Value.

Furthermore, the standard deviation varies according to distance. The sensor's and the gateway's average standard deviations were 4.30 and 3.20, respectively.

5.5.5 Downlink Sensor RSSI and Gateway RSSI with Curve Fitting

Curve fitting is a mathematical technique used to find a line or curve that best represents a set of data points. The main objective of curve fitting is to create a model that accurately represents the data and can be used to make predictions or analyze trends.

There are several methods of curve fitting, including linear regression, polynomial regression, and non-linear regression. Linear regression involves finding the best-fit line that represents the data points. Polynomial regression involves finding the best-fit curve that represents the data points. Non-linear regression is used when the relationship between the independent and dependent variables is not linear. In this project, the non-linear regression curve fitting is used. The least squares method is a popular approach in curve fitting. It is used to find the best-fitting curve that minimizes the sum of the squared residuals between the observed data points and the estimated curve. The least squire method is also used in this project. The following code was used to calculate the reference received signal strength (A) and signal propagation constant (n).

Below is a snippet of the code utilized for implementing the curve fitting process. This code segment is a fundamental component to determination of essential parameters for optimizing the system's performance.

```
from scipy.optimize import curve_fit
x = sensor_gateway['Distance (meter)'].iloc[:-1]
y = sensor_gateway['Sensor RSSI (dBm)'].iloc[:-1]
yy = sensor_gateway['Gateway RSSI (dBm)'].iloc[:-1]

def f(x, A, n):
    return (-A-10*n*np.log10(x))

def ff(x, A, n):
    return (-A-10*n*np.log10(x))

par, var = curve_fit(f, x, y)
par2, var2 = curve_fit(ff, x, yy)

A, n = par
print('A:',A)
print('n:', n)

A2, n2 = par2
print('A2:',A2)
print('n2:', n2)
```

Figure 53 Curve Fitting python code

Utilizing the curve fitting technique, we derived the reference received signal strength (A) and signal propagation constant (n) values for both the sensor and the gateway. The calculated values are as follows: -55 dBm and 7.5 for the sensor, and -18 dBm and 6.3 for the gateway, respectively. These parameters play a critical role in optimizing signal transmission and reception, ensuring robust and efficient communication between devices.

The x-axis corresponds to the independent variable of distance, represented on a logarithmic scale, whereas the y-axis represents the dependent variable RSSI in dBm. This logarithmic representation of distance allows for a more comprehensive analysis of the relationship between distance and RSSI, offering valuable insights into signal propagation characteristics and signal strength variations over varying distances.

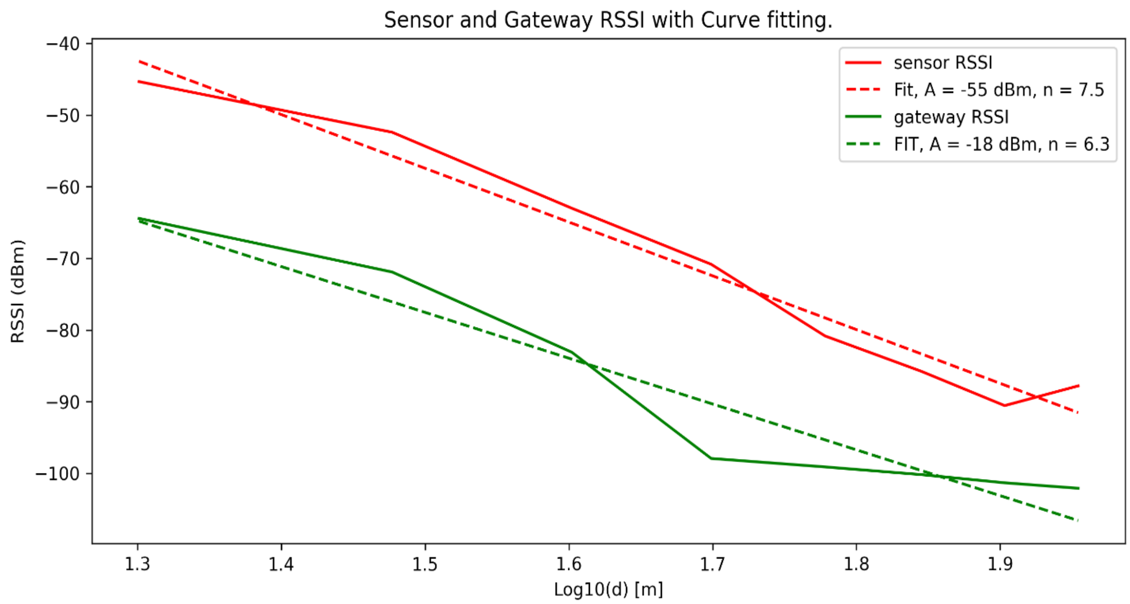

Figure 54 Sensor and Gateway RSSI line with their Curve fitting line in Logarithmic Scale.

In the case where the x-axis represents distance as independent variables on a linear scale, and the y-axis represents RSSI as dependent variables in dBm, the graph is presented below.

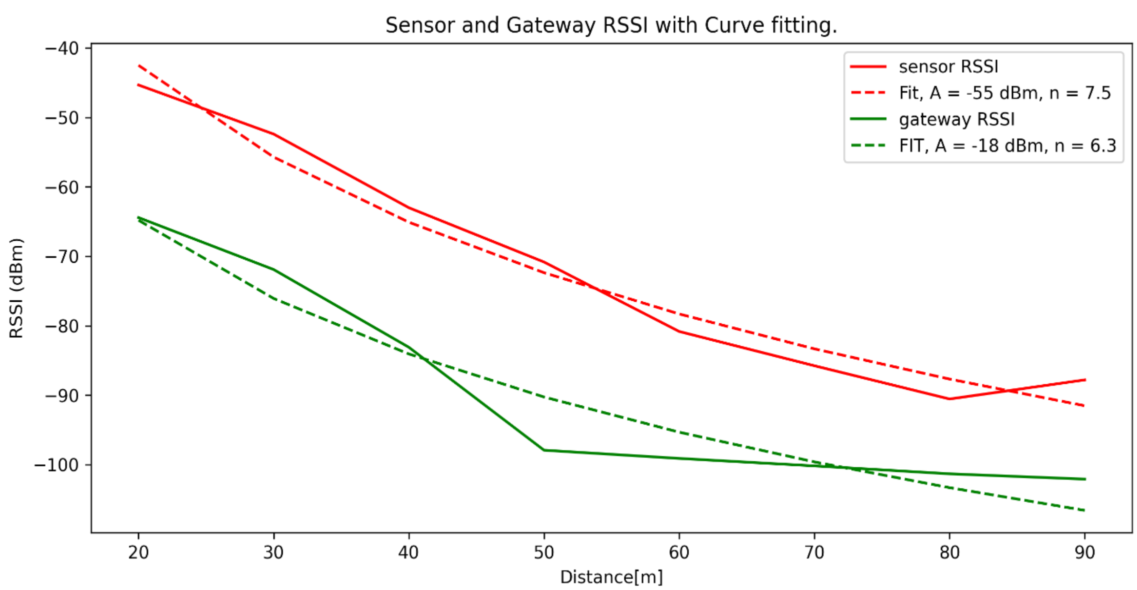

Figure 55 Sensor and Gateway RSSI line with their Curve fitting line in Linear Scale.

5.5.6 Short and Long Antennas at RN2483 Module

The RN2483 module is equipped with both short and long antennas, offering flexibility in terms of range and signal strength. This allows for optimal performance in various applications. The small antenna is suitable for short-range communication[26], making it ideal for applications where space is limited. On the other hand, the large antenna provides a longer range[26] and stronger signal, making it suitable for applications that require communication over longer distances. The module's ability to switch between these two antennas gives users the freedom to choose the most appropriate option based on their specific needs.

Within this project, RSSI values were evaluated at two separate locations, revealing that the RSSI values exhibited minimal variance between short and long antennas. The difference between them approximately 4 dBm to 5 dBm. This observation suggests that, in these particular environments, antenna length had a limited impact on signal strength. Such findings contribute valuable insights into the suitability of antenna options for specific deployment scenarios, potentially simplifying antenna selection considerations for similar projects.

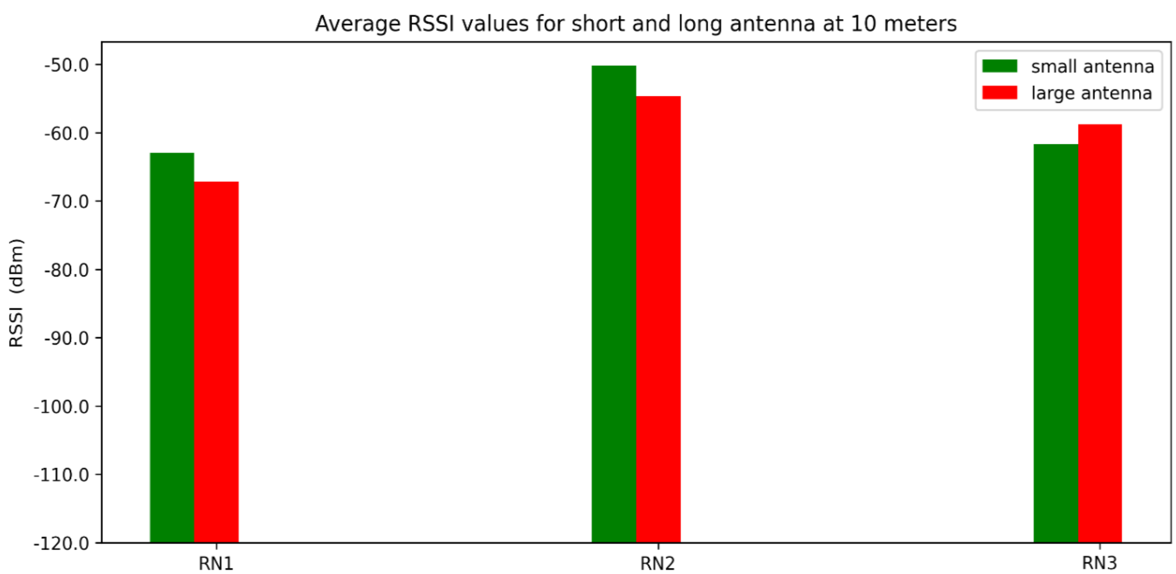

Figure 56 Average RSSI value for Short and Long Antenna in 10 meters distance for RN2483 Module.

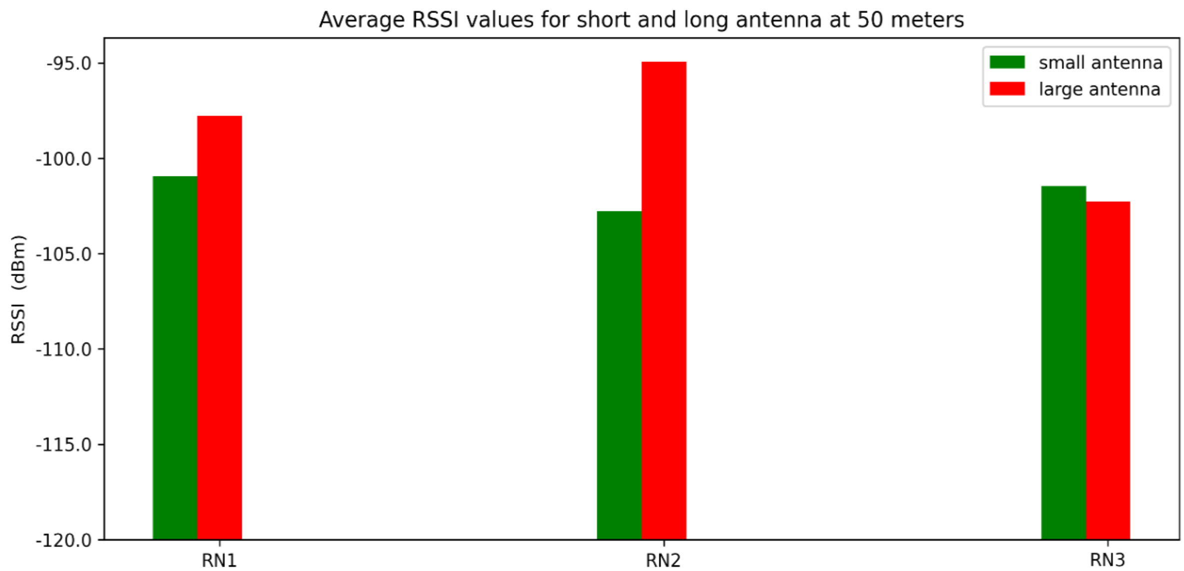

Figure 57 Average RSSI value for Short and Long Antenna in 50 meters distance for RN2483 Module.

5.6 Percentage of Lost Packets

The percentage of lost packets in a gateway is a significant indicator that indicates network communication reliability and efficiency. When data packets are transmitted from one device to another through a gateway, there is a possibility of some packets getting lost or dropped along the way. This can be due to various factors such as network congestion, hardware issues, or software errors. High packet loss rates can lead to degraded network connectivity, slow data transfer speeds, and increased latency.

	green	LA66_66	LA66_35	LA66_45	LHT65_91	LHT65_27A	LHT65_86	USB_2	USB_3	USB_72F	RN_1	RN_2	RN_3	RN_4	RN_5_PCB
10	0	0	2	2	4	0	0	6	0	0	0	0	0	0	0
20	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
30	0	3	1	1	0	1	0	0	0	2	0	0	0	0	0
40	0	1	1	1	0	6	0	0	0	10	0	9	0	0	0
50	0	1	0	0	0	0	1	0	0	0	0	15	0	0	0
60	0	0	2	0	2	0	3	0	0	0	0	1	0	1	0
70	0	1	4	3	0	0	0	0	0	0	0	8	0	0	0
80	0	0	0	0	15	4	4	1	0	1	4	3	0	0	0
90	0	0	1	1	14	3	2	0	1	3	0	9	4	0	0

Table 10 Number of Packet Loss in Various Positions for Different Sensors.

As indicated by the information presented in the above table, there is a direct correlation between distance and the rate of lost packets, as distance extends, the rate of packet loss proportionally rises. Furthermore, the data table highlights disparities in packet loss rates among various sensors. Notably, LHT 65 sensor exhibits a notably higher rate of packet loss, while in contrast, the RN2483 sensor demonstrates a considerably lower incidence of packet loss. Among all the sensors tested, the RN2483_pcb stands out as the most reliable, exhibiting zero packet loss across varying distances. This observation highlights the exceptional performance of the RN2483_pcb sensor in maintaining data integrity, which is crucial in applications where data consistency and reliability are important. These findings underscore the significance of sensor selection and placement in network design, as they directly impact the robustness and reliability of data transmission over varying distances. Effectively managing packet loss is pivotal for optimizing communication performance and data integrity.

The bar graph below represents the percentage of lost packets for various sensors located at different points. The graph's x-axis represents the different signals being measured at various positions, while the y-axis represents the percentage of lost packets.

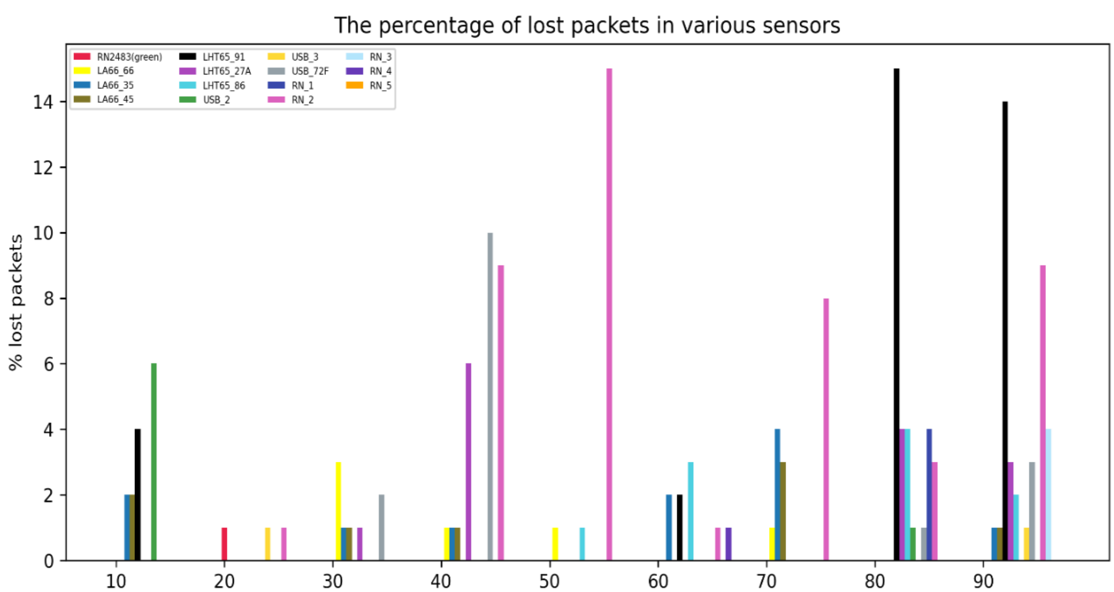

Figure 58 The Number of packets lost at various positions.

5.7 Issue During Join to the Gateway

I encountered two distinct issues when trying to connect more sensors to the gateway in my project: "access denied" and "channel busy."

Access Denied: This error message indicates that the gateway is rejecting the sensor's connection request. It could be due to security settings, or compatibility issues.

Channel Busy: The "channel busy" message typically occurs when the communication channel used by the gateway is occupied or experiencing interference, making it challenging for the sensor to transmit data.

The RN2483 and LA66 LoRaWAN modules operate at distinct baud rates. It's crucial to ensure that the baud rate settings match; otherwise, the sensor won't be able to establish a connection with the gateway. Specifically, the default baud rate for the RN2483 is set at 57600[11], whereas the LA66 LoRaWAN module operates at a default baud rate of 9600[10]. Properly configuring the baud rate is essential for seamless communication between the sensors and the gateway.

When attempting to establish a connection with the RN2483 module, I encountered difficulties in establishing a connection with the gateway. This issue became more pronounced when a greater number of sensors were simultaneously connected to the gateway. It's essential to note that the number of connected sensors can impact network performance and may require adjustments in settings or configurations to ensure seamless connectivity.

If the sensors fail to establish a connection with the gateway, it is recommended to disconnect the sensors from the USB port, then reconnect them, and attempt to initiate the joining process with the gateway once more. This step can help resolve any potential communication issues and facilitate the successful integration of the sensors into the gateway.

After successfully establishing a connection with the gateway, the connection remains consistently reliable and steady. This stability is crucial for uninterrupted data transmission and ensures that the devices can communicate effectively with the gateway throughout the operation

6 Conclusion

In this master's thesis project, the focus was on utilizing Arduino, STM32 Nucleo microcontrollers, and LoRaWAN technology for transmitting sensor data to the ThingsNetwork. The project began by connecting the LA66 LoRaWAN module to a gateway and sending data using AT commands through serial communication. Subsequently, the LA66 module was integrated with the Nucleo Microcontroller to transmit temperature and humidity data.

As the project progressed, various sensors, including LA66 LoRaWAN USB, RN2483, and LHT65, were added to the gateway, leading to an investigation of sensor joining and packet loss issues. Notably, shorter transmission intervals (less than 10 seconds) were found to result in increased packet loss, prompting the adoption of a 10-second interval to mitigate this issue.

The project extended to connect a total of 15 different sensor types to the gateway, revealing a correlation between sensor expansion and increased packet loss during the transmission. The research focused into the evaluation of RSSI (Received Signal Strength Indicator) values, which demonstrated an inverse relationship with distance, indicating stronger RSSI values for sensors closer to the gateway and weaker values for those at greater distances.

Another key aspect explored in the study was packet loss, and the results suggested that it increased as the sensor-to-gateway distance extended.Among the tested sensors, the LHT 65 sensor exhibited higher packet loss rates, while the RN2483 sensor displayed greater reliability, experiencing zero packet loss across varying distances.

The research also involved the calculation of the Signal Propagation Constant (n), which was determined to be 5, indicating indoor signal propagation[25]. Additionally, lower Reference Received Signal Strength (A) values were associated with lower packet loss, while higher A values correlated with increased packet loss.

Furthermore, the project explored antenna performance with the RN2483 module, finding no significant difference in performance between two types of antennas. Lastly, a notable observation was made regarding the difference between Sensor RSSI and Gateway RSSI, which amounted to approximately 15 dBm.

In conclusion, this master's thesis provided valuable insights into optimizing LoRaWAN communication networks. Monitoring RSSI values and understanding the signal propagation constant were identified as strategies to enhance communication range and performance. The project underscored the challenges of expanding sensor connectivity, as it may lead to sensor joining issues, potentially disrupting network functionality with gateways.

Future Research

Further research can be conducted in different areas. Firstly, it involves investigating the effects of changing the spreading factor, which can significantly influence communication performance. Secondly, comprehensive testing in diverse environmental conditions. Thirdly, examining Signal-to-Noise Ratio (SNR) will help in understanding the quality of signal reception and its implications for communication reliability. Lastly, the study will also explore the correlation between payload size and the efficiency of communication. These areas of research promise to enhance our understanding of IoT communication and contribute to more robust and efficient sensor networks.

Key Points:

RSSI (Received Signal Strength Indicator), Signal Propagation Constant, LA66 LoRaWAN, RN2483, LHT65, Distance, LoRa Gateway, Moving Average, Attenuation, Access Denied, Channel busy.

7 Bibliography:

[1] N. H. Motlagh *et al.*, ‘Digital Twins for Smart Spaces - Beyond IoT Analytics’, *IEEE Internet Things J.*, pp. 1–1, 2023, doi: 10.1109/JIOT.2023.3287032.

[2] J. R. Santana, P. Sotres, J. Pérez, L. Sánchez, J. Lanza, and L. Muñoz, ‘Assessing LoRaWAN radio propagation for smart parking service: An experimental study’, *Computer Networks*, vol. 235, p. 109962, Nov. 2023, doi: 10.1016/j.comnet.2023.109962.

[3] S. Sadowski and P. Spachos, ‘RSSI-Based Indoor Localization With the Internet of Things’, *IEEE Access*, vol. 6, pp. 30149–30161, 2018, doi: 10.1109/ACCESS.2018.2843325.

[4] M. A. Ertürk, M. A. Aydın, M. T. Büyükakkaşlar, and H. Evirgen, ‘A Survey on LoRaWAN Architecture, Protocol and Technologies’, *Future Internet*, vol. 11, no. 10, p. 216, Oct. 2019, doi: 10.3390/fi11100216.

[5] A. Lavric and V. Popa, ‘Internet of Things and LoRa™ Low-Power Wide-Area Networks: A survey’, in *2017 International Symposium on Signals, Circuits and Systems (ISSCS)*, Iasi, Romania: IEEE, Jul. 2017, pp. 1–5. doi: 10.1109/ISSCS.2017.8034915.

[6] E. D. Ayele, C. Hakkenberg, J. P. Meijers, K. Zhang, N. Meratnia, and P. J. M. Havinga, ‘Performance analysis of LoRa radio for an indoor IoT applications’, in *2017 International Conference on Internet of Things for the Global Community (IoTGC)*, Funchal: IEEE, Jul. 2017, pp. 1–8. doi: 10.1109/IoTGC.2017.8008973.

[7] H. Kwasme and S. Ekin, ‘RSSI-Based Localization Using LoRaWAN Technology’, *IEEE Access*, vol. 7, pp. 99856–99866, 2019, doi: 10.1109/ACCESS.2019.2929212.

[8] M. Aref and A. Sikora, ‘Free space range measurements with Semtech Lora™ technology’, in *2014 2nd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems*, Odessa, Ukraine: IEEE, Sep. 2014, pp. 19–23. doi: 10.1109/IDAACS-SWS.2014.6954616.

[9] J. Grübel *et al.*, ‘Dense Indoor Sensor Networks: Towards passively sensing human presence with LoRaWAN’, *Pervasive and Mobile Computing*, vol. 84, p. 101640, Aug. 2022, doi: 10.1016/j.pmcj.2022.101640.

[10] ‘Dragino LA66 AT Command Sets’. Dragino LoRa® AT Command Sets, 2022. [Online]. Available: https://www.dragino.com/downloads/downloads/LT_LoRa_IO_Controller/DRAGINO_LT_AT%20Commands_v1.1.pdf

[11] Microchip Technology Inc., ‘RN2483 LoRa® Technology Module Command Reference User’s Guide’. 2018 2015. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/RN2483-LoRa-Technology-Module-Command-Reference-User-Guide-DS40001784G.pdf

[12] Dragino, ‘Dragino LHT65 AT Command Sets’. 2019. [Online]. Available: www.dragino.com

[13] Dragino, ‘LHT65 Temperature & Humidity Sensor User Manual’.

[14] Microchip Technology Inc., ‘MCP1700’. [Online]. Available: https://ww1.microchip.com/downloads/aemDocuments/documents/APID/ProductDocuments/DataSheets/MCP1700-Data-Sheet-20001826F.pdf

[15] ECE department, IUPUI, Indianapolis, USA. *et al.*, ‘Building a Private Lo Ra WAN Platform’, *IJEAT*, vol. 9, no. 1s3, pp. 291–295, Dec. 2019, doi: 10.35940/ijeat.A1055.1291S319.

[16] ‘SHT31 Weather-proof Temperature & Humidity Sensor’. https://www.mouser.de/new/dfrobot/dfrobot-sht31-temperature-humidity-sensor/?_gl=1*1pjftyx*_ga*dW5kZWZpbmVk*_ga_15W4STQT4T*dW5kZWZpbmVk*_ga_1KQLCYKRX3*dW5kZWZpbmVk
[17] J. Cannon *et al.*, ‘Applications of low-cost environmental monitoring systems for fine-scale abiotic measurements in forest ecology’, *Agricultural and Forest Meteorology*, vol. 321, p. 108973, Jun. 2022, doi: 10.1016/j.agrformet.2022.108973.
[18] STMicroelectronics, ‘STM32 Nucleo-32 boards’. [Online]. Available: https://www.st.com/resource/en/data_brief/nucleo-l031k6.pdf
[19] Dragino, ‘LA66 LoRaWAN Module’. [Online]. Available: https://www.dropbox.com/sh/z6daguv1gach8ce/AAAw_FjksNXD91bE3aJOOmnta?dl=0&preview=Datasheet_LA66+LoRaWAN+Module.pdf
[20] Microchip Technology, ‘Low-Power Long Range LoRa® Technology Transceiver Module’. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/50002346C.pdf
[21] Dragino, ‘LHT65 LoRaWAN Temperature & Humidity Sensor’. [Online]. Available: https://www.dragino.com/downloads/downloads/LHT65/UserManual/DRAGINO_LHT65_AT_Commands_v1.0.pdf
[22] Christian A. Hviid, Anders Struck, and Kevin Michael Smith, ‘A cost-effective and versatile sensor data platform for monitoring and analysis of building services’, aivc. [Online]. Available: https://www.aivc.org/sites/default/files/D2_S8A-04.pdf
[23] K. Mikhaylov and J. Tervonen, ‘Evaluation of Power Efficiency for Digital Serial Interfaces of Microcontrollers’, in *2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*, Istanbul, Turkey: IEEE, May 2012, pp. 1–5. doi: 10.1109/NTMS.2012.6208716.
[24] P. Timtere, Y. Ibrahim and D. Yabwa, ‘Correlation Between Radio Signal Strength and Attenuation as a Function of Linear Distance and Time’, 2018. doi: 5829/idosi.wasj.2018.434.440.
[25] J. Miranda *et al.*, ‘Path loss exponent analysis in Wireless Sensor Networks: Experimental evaluation’, in *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, Bochum, Germany: IEEE, Jul. 2013, pp. 54–58. doi: 10.1109/INDIN.2013.6622857.
[26] Tapan Pattnayak, Guhapriyan Thanikachalam, ‘Antenna Design and RF Layout Guidelines’. [Online]. Available: http://www.cypress.com/go/AN91445

Appendix

Keys:

The DevEUI, AppEUI, and AppKey identifiers play a crucial role in the implementation of LoraWAN technology. These identifiers are essential for establishing secure and efficient communication between devices in a LoraWAN network.

The DevEUI, which stands for Device EUI, is a unique identifier assigned to each device in the network. It is a 64-bit value that is globally unique and cannot be modified. The DevEUI allows network servers to identify and differentiate between different devices in the network.

AppEUI, or Application EUI, is another important identifier used in LoraWAN. Like the DevEUI, it is also a 64-bit value that is globally unique. The AppEUI is used to identify the application or service to which a device belongs. It helps in routing messages to the correct destination within the LoraWAN network.

AppKey, on the other hand, is a 128-bit key that serves as an encryption key for securing communication between devices and application servers. Each device in the LoraWAN network has its own unique AppKey. It is used to encrypt and decrypt data exchanged between the device and the application server, ensuring data privacy and security.

The table below represents the various keys for various sensors.

- **LA66 LoRaWAN Module (735)**

In this project, LA66 LoraWAN module is used which is highly versatile and can be easily integrated into various IoT devices. It supports multiple communication interfaces, including UART and SPI, making it compatible with a wide range of microcontrollers and sensors. The table below contains all the different key information for this module.

AppEUI	A840410000000101
DevEUI	A840418DD184D735
AppKey	1C392C6FC9B8F73332A8D29978C944C2
Device address	260B03DD
NwkSKey	6017622E76596E65A68FF425A56A3F93
AppSKey	E9AE4D318ED1FCC11BD5F88A475D02C1
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 11 : LA66 LoRaWAN module key (735) values

AppEUI	A840410000000101
DevEUI	A84041EB8184D745
AppKey	3C4EFB7C1468CA1CCEBF19EC4BC78412
Device address	260BBCC5
NwkSKey	4CC7AC2C8A38D1AA5A36C92D5C149288
AppSKey	EB67EBD9D0F1736395619F1123C873F7
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 12 LA66 LoRaWAN module key (745) values

AppEUI	A840410000000101
DevEUI	A84041D11184D97D
AppKey	FCC674A556AA89BEA6EBCD7BFD463264
Device address	260B70C7
NwkSKey	7842C5B51D9F283678BEDB43F9EE250D
AppSKey	B0292AD9B9156F7F60A9D75D6124DF91
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 13 LA66 LoRaWAN module key (66) values

AppEUI	A840410000000101
DevEUI	A84041948184D72F
AppKey	395B877D952A87EB87B3C62EF9E3FC8C
Device address	260BCC5B
NwkSKey	F0F921FAEF83988DCAEB728B0347DF63
AppSKey	7A1E84A0EA5572A5879F4EB77F98FAF2
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 14 LA66 LoRaWAN USB -1 key and values

AppEUI	A840410000000101
DevEUI	A84041BF3184D993
AppKey	CA2C91AA2BCFDC112AADFD5E52FD6AFD
Device address	260B7877
NwkSKey	5D2EA9C9A5E3FE793B725F3CFE3EABFD
AppSKey	CCFCCF298E1A0450694728007D34403B
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 15 LA66 LoRaWAN USB -2 key and values

AppEUI	A840410000000101
DevEUI	A8404172D184D98D
AppKey	3B1992985BDECEE3AC1BF6B4FE69141E
Device address	260B8E28
NwkSKey	69CA7CF06356A4CCAB1E94FA368A82D3
AppSKey	25007EA32E271BE48835C78D587C3601
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 16 LA66 LoRaWAN USB -3 key and values

❖ Some Useful AT commands for LA66 LoRaWAN Module:

AT commands are used to configure and control the LA66 LoRaWAN module. However, the specific AT command set can vary depending on the firmware and configuration of the module. It's essential to refer to the module's datasheet, user manual supported by your LA66 module. Below are some important common AT commands

1. **AT**: Test command to check if the module is responsive.
2. **AT+VERSION**: Returns the firmware version of the module.
3. **AT+RESET**: Resets the module.
4. **AT+CFG**: Print all configurations.
5. **AT+JOIN:** Join to the Network.
6. **AT+RSSI?** : Get the RSSI of the last received packet.
7. **AT+SENDB=hexdata**: Sends hexadecimal data along with application port on the LoRaWAN network.

Example of send data on the LoRaWAN network.

AT+SENDB = 0, 3, 4, 11223344

- **LHT65 sensor**

LHT65 sensors can collect data on temperature, humidity, and pressure, making them versatile and adaptable to different environments. In addition to their long battery life, LHT65 sensors also offer excellent connectivity options.The table below contains all of the module's key information.

AppEUI	A000000000000100
DevEUI	A84041C83183527A
AppKey	F616FF6CAE55BC213513D7F43F8D9C36
Device address	260BA0F6
NwkSKey	69B7BE4A52FCE151D1902123E568CE74
AppSKey	E548CE85D15717650062C487E94FF824
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 17 : LHT 65 (27A) sensor key values.

AppEUI	A000000000000100
DevEUI	A840418D71835286
AppKey	E32598D689FDED822993C2ABB457912A
Device address	260BE96A
NwkSKey	93C71C025F0E43FA67EF67A28D7412D5
AppSKey	241270D51DF9E8D0C8F1DF87B6165752
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 18 LHT 65 (86) sensor key values.

AppEUI	A000000000000100
DevEUI	A8404194B1835291
AppKey	B79763B42EBB18694875B563ED5FAF1C
Device address	260B6A42
NwkSKey	524DFF2B09DD0FDF7B49CF61021B49F7
AppSKey	5D224A4A6C3A4B73B8A61B5480E604BD
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 19 LHT 65 (91) sensor key values.

- **RN2483 Module**

The RN2483 Module is a device that enables long-range wireless communication using the LoRa technology. It is designed for low-power applications and operates in the 868 MHz frequency band. The table below shows the key for this module.

AppEUI	70B3D57ED002F9AD
DevEUI	0004A30B001B864E
AppKey	A6698AC27575A5BCA56869858510499C
Device address	260BCCE8
NwkSKey	7EFAAE0CAC921C53CD7C6A08E133CA9C
AppSKey	22E722D033D9B3D5AB9200D674015A32
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 20 RN2483 (Green) Modules key values.

AppEUI	FEDCBA9876541230
DevEUI	70B3D57ED005F9F3
AppKey	3FAE32AFD725470A59E92046B0A7026D
Device address	260BBA0C
NwkSKey	F58162F8231E26C73E1E08FA6238C05E
AppSKey	9D4FB81B25DE2AD3B1AAC294393868D6
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 21 RN2483 (Yellow) Modules key values.

AppEUI	FEDCBA9876543211
DevEUI	70B3D57ED005C4BD
AppKey	8FC3AC798A7A27189F5138F40A9E43FF
Device address	260B31B5
NwkSKey	063EF8A69791BBCD829A20E09A5BA9DD
AppSKey	D28DA0ADACFF23E2AE5DB6B9C61C84B7
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A

Table 22 Soldering RN2483 (1) key and values.

AppEUI	FEDCBA9876549210
DevEUI	70B3D57ED00603BF
AppKey	889FB5FF6739D6EFC71CAEA55ADD2FF5
Device address	260B23EA
NwkSKey	6076AD64C238F29C48B8477C3867DF83
AppSKey	19A8DB4FB5AD8266978AB8556283FFBB
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.1
Regional Parameters version	TS001 Technical Specification 1.0.1

Table 23 Soldering RN2483(2) key and values.

AppEUI	FEDCBA9876549910
DevEUI	70B3D57ED00603C3
AppKey	9C4C873EB2F67488FA9ED68431569959
Device address	260BBFDB
NwkSKey	D78B74E8FB60FF6785A59AEA932614FD
AppSKey	D1804D1274357C6D561A95C31A17B7BE
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.1
Regional Parameters version	TS001 Technical Specification 1.0.1

Table 24 Soldering RN2483(3) key and values.

AppEUI	FEDCBA9876549990
DevEUI	70B3D57ED00603C6
AppKey	7EE0FC0F876C33173EB6DF25758391B8
Device address	260B85FD
NwkSKey	1B79807C5D566CCAF754525D72F581A1
AppSKey	13578BA8EFF83AAC13A40C210A23AF8A
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommended
LoRaWAN version	LoRaWAN Specification 1.0.1
Regional Parameters version	TS001 Technical Specification 1.0.1

Table 25 Soldering RN2483(4) key and values.

❖ Some Useful AT commands for RN2483 Module:

The RN2483 is a popular LoRa (Long-Range) transceiver module from Microchip Technology. Here are some common AT commands for configuring and communicating with the RN2483 module:

1. **sys reset**: Resets and restarts the RN2483 module.
2. **sys get hweui:** Read the preprogrammed EUI Node address.

3. **mac join <mode>:** Attempt to join the network.
   <mode>: Over-The-Air Activation (OTAA) or Activation By Personalization (ABP)

4. **mac tx <type> <portno> <data>:** Send data to the Gateway.
   <type>: String represent the uplink payload type, either **cnf -** confirmed or **uncnf –** unconfirmed.
   <portno>: Represent the port number from 1 to 223.
   <data>: Hexadecimal value. The length of data bytes capable of being transmitted are dependent upon the set data rate.

Example of send data on the Gateway.

mac tx cnf 4 11223344

5. **mac save**

- **Gateway**

The LoRaWAN Gateway is a crucial component in a LoRaWAN network, serving as the bridge between end-devices and the network server.

Additionally, the gateway manages the encryption keys used for secure communication. These keys are used to secure the communication between end-devices and the network server, ensuring that data remains private and protected from unauthorized access.

Gateway ID	dca632fffede8d59
Gateway EUI	DCA632FFFEDE8D59
Frequency plan	EU_863_870_TTN

Table 26 Gateway keys values

- **Code Structure**

In this project, various modules are employed, with each module utilizing distinct programming code for execution. The code's structure offers a well-organized framework, simplifying both navigation and maintenance tasks. The figure illustrates the structure of the code.

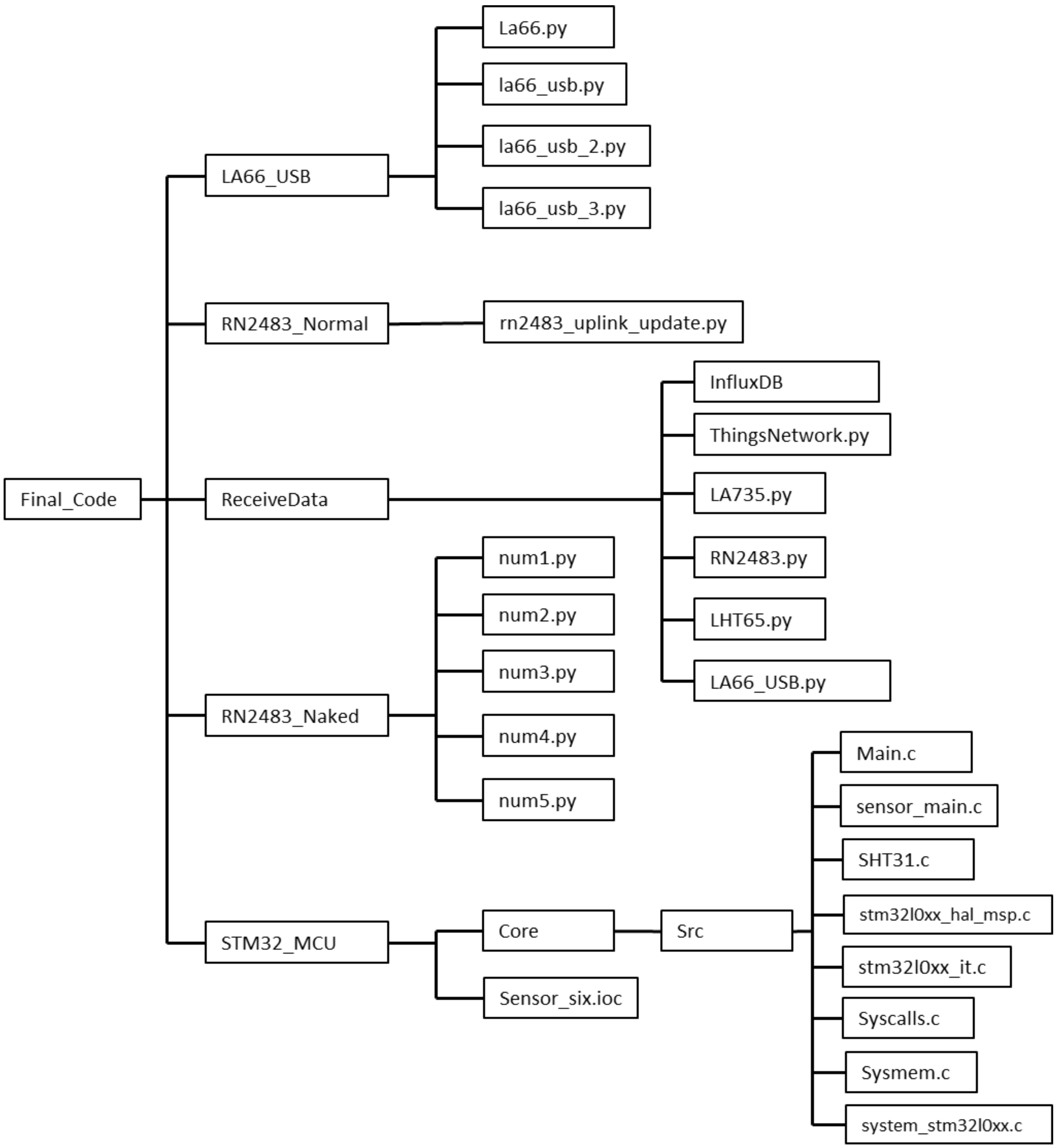

Figure 59 Code Structure in SVN.

- **Code Explanation**

Below is a list of the Python libraries used in this project.

- Paho.mqtt
- Json
- Pandas
- Numpy
- Scipy
- Pyserial
- Argparse
- Os
- InfluxDB
- Matplotlib
- Seaborn

LA66_USB: The code contained within this directory is associated with LA66 LoRaWAN USB. This code serves the purpose of establishing communication with the LA66 LoRaWAN USB module. The programming code is used to send the data to the ThingsNetwork.

- ***la66.py/ la66_usb.py/ la66_usb_2.py/ la66_usb_3.py:*** These programs are designed to read data from a serial port and can be run on a laptop or a Raspberry Pi. It's important to ensure that you correctly specify the port number for each individual module before running the code. This is typically done to establish a connection with the specific device or module you want to communicate with via the serial port. The port number is essential to identify and establish the communication link with the targeted device.

RN2483_Normal: This directory is associated with programming code related to the RN2483 (green) module, which is utilized for transmitting data to the ThingsNetwork.

- ***rn2483_uplink_update.py:*** This program is used to be run on either a laptop or a Raspberry Pi, and its purpose is to establish serial communication with an RN2483 module. The code specifies a baud rate of 57600 for the serial communication, and it's crucial to ensure that the correct port number is configured to establish a successful connection between the computer (laptop or Raspberry Pi) and the RN2483 module.

ReceiveData: This directory contains the relevant programming code for receiving messages from the ThingsNetwork.

- ***ThingsNetwork.py****:* The 'ThingsNetwork.py' script is used to retrieve data from the ThingsNetwork. This program is executed on a laptop. Once the message is received, it filters out the relevant information and stores it in a file.
- ***InfluxDB: The*** InfluxDB folder contains code designed for sending important data to the InfluxDB database. To enable data transmission, it is essential to configure the ServerConfig.py file.

RN2483_Naked: This directory contains programming code related to the RN2483(Naked) module which is used for data transmission to the ThingsNetwork. This program is run as like RN2483_Normal.

STM32_MCU: This directory contains the relevant programming code for the STM32 microcontroller which is written in C. This code serves the dual purpose of transmitting data to the ThingsNetwork as well as handling downlink transmission.

- ***main.c:*** This is the main function which initiates and controls the execution of other modules or functions based on specific requirements or conditions within the program.
- ***sensor_main.c:*** This program is utilized to enable the I2C communication protocol and retrieve temperature and humidity data from the SHT31 sensor.
- ***SHT31.c:*** This programming code is employed for the conversion of raw temperature and humidity data into human-readable values. Additionally, it facilitates the process of downlink transmission of this data.