IoT-based Data Collection and Transmission for Thermal Water Supply Systems in a Smart Home: Implementation and Testing

STIAN SCHEDIN HALVORSEN

1

SUPERVISORS
Geir Jevne
Frank Yong Li

University of Agder, 2021
Faculty of Engineering and Science

Abstract

This thesis will look into theoretical aspect, implementation and design of data collection and data communication systems. The core system that this thesis use as a data collection system is a thermal water system that exploit heat exchange from a well. Most problems that occur in heat transfer systems can be avoided by implement a automatic surveillance system. This thesis will review basic functionality of the intended thermal system and operative problems and further propose a surveillance system using theory in data collection and data transmission- To transfer the data collected from the thermal system a communication protocol called Message Queuing Telemetry Transport(MQTT) is used. MQTT is an OASIS standard messaging protocol and is designed as an lightweight publish/subscribe messaging transport. MQTT is used in a wide variety of use cases within different sectors to automate control systems. Further this thesis will also provide insights in a relatively new LPWAN protocol called 802.11ah (Wi-Fi HaLow) and how it will deal with the requirements for the Internet of Things (IoT) and smart environments. This include looking into functionality and existing research, implementation, and testing of the protocol to analyze performance.

Contents

List of Figures

Abbreviations

ADC Analog-to-Digital Converter

DSI Display Serial Interface

EMS Electromagnetic Spectrum

ERP Effective Radiated Power

GUI Graphical User Interface

I2C Inter-Integrated Circuit

IEEE Institute of Electrical and Electronics Engineers

IDE integrated development environment

International Electrotechnical Commission IEC

IoT Internet of Things

ISP Internet Service Provider

LPWAN Low Power Wide Area Network

LWT Last Will Testament

M2M Machine-to-Machine

MSC Modulation Coding Scheme

MTC Machine Type Communication

MQTT Message Queuing Telemetry Transport

NKOM National Communication Authority (Nasjonal KommunikasjonsMyndighet)

OS Operative System

OSI Open Systems Interconnection

PMR Private Mobile Radio

QoE Quality of Experience

QoL Quality of Life

QoS Quality of Service

RAW Restricted Access Window

RPi Raspberry Pi

RSSI Received Signal Strength Indicator

RTD Resistance Temperature Detector

SNR Signal To Noise Ratio

SoC System on a Chip

SRD Short Range Device

TCP Transmission Control Protocol

TLS Transport Layer Security

TWT Target Wake Time

Chapter 1

Introduction

1.1 Background and Motivation

Our society is in constant change and new ideas is being presented regularly to improve the existing systems or bring new innovated solutions to a specific problem. During the last decades the Internet of Things (IoT) phenomenon has emerge and stands towards technical challenges such as power consumption, speed, spectrum occupation, range restrictions and more. Many possible solutions have been proposed to solve these challenges in the format of different protocols. Still, they can always be enhanced to make the systems prepared for new applications and requirements.
One of the most common IoT usage is the private home section where devices collaborate together that provides the house with automated functions and operations for the user based on either personal preference or needed surveillance.

Through the educational process it has present our society in a complex composition of data systems with several technologies and protocols to comprehends the massive information system the earth has become. Through the perspective it has been in interest to test and analyze a full developed data system in a practical life environment to understand the different concepts within data information systems.

In the practical execution of this project it is in interested to be able to present the different sections within a data collection and data transmission system to further help comprehend the complexity of such systems. Thus making it easier to evaluate and understand the overall structure of information systems.

1.2 Objective

Research questions/Project goals: This thesis is divided into three parts to cover the basics of data handling and data communication. The first objective is to develop design and implement a sensor system based on existing software and hardware to present the basics within data collection. The second objective is to look into data transmission and further integrate a suggested communication protocol to objective one. The final objective and where the research question is presented is where a relatively new protocol is being presented to improve and solve technical challenges within sensor and data networks.

1.2.1 Data Collection for Thermal Water Supply System

The project objective is to design and implement a smart home system software with all the necessary components for data handling. The home environment system zone shall gather information from a thermal water supply system, this information shall be used to state the efficiency of the system as well bring operational data to the user through a GUI.

1.2.2 Data Transmission Based on MQTT

The project will also look into the MQTT protocol with the collaboration of the thermal water supply system to transfer the data through the central hub and the module that collects the data. The protocol shall collaborate with the data collection software for the thermal water supply system that will take into account user configuration and the system critical structure.

1.2.3 Overview and Test of 802.11ah

The project research question is towards the relatively new Wi-Fi protocol 802.11ah Wi-Fi HaLow. The project will present a general overview of the protocol as well as test the protocol with relevant variables that can affect the theoretical specifications and present the analytic data based on the performed test. Research question: How will 802.11ah technical specifications perform in a realistic environment in accordance with Norwegian law and restrictions?

1.3 Methodology

To complete this project it is not without taking into account not only technical challenges but also public measures to complete a fully developed information system. Below is described the theoretical aspect covering the different influenced sections of this thesis.

1.3.1 Laws and Regulations Survey

Part of this project will perform testing of a communication protocol that is not implemented in Norwegian official marked, which means all laws and regulations have to be checked in correspond with the electronic device spectrum usage. The first thing to focus on is to get spectrum permission in the frequency scope of the electronic device that is being used to test the protocol.

1.3.2 Literature Review

The theoretical basis for practical develop a home system is based on online guides from Raspberry Pi official website. For research regarding the theoretical aspects of IoT and Smart homes, online articles from IEEE, earlier lectures and publications in ICT from University of Agder is used.

Overview of the many protocol and technology collaborations is important to understand the concept of smart homes and IoT. A survey from IEEE done in 2015 is explaining the progress and growth of the IoT concept, including enabling technologies, protocols, and application issues [2]. This survey presents the general concept and structure of IoT:

- Architecture with possible explanations towards layer structure.
- Its functionality is explained through six elements.
- A overview of common prominent protocols used in the IoT environment at different layers.
- IoT requirements, Challenges, and Future Directions.
- Overview of support systems for IoT, such as clouds and big data analytics.

An important factor which is missing is the more deeply analytics regarding cost and implementation through a customer set of perspective and/or a practical execution of an IoT system.

Regarding the project smart home zone, it required some theory to understand the application functionality and needed surveillance. A report from Sintef research institute has a report from its energy research department which focuses on heat pumps and building heating [11]. Sintef‘s report creates a description of the application construction, functionality, and cautions during its operative state, and will then form a basis for the project smart home zone application.
Sintef‘s report forms a general construction and explanation of a heat pump system, what missing from this report is an updated version of surveillance and control systems. This will not be a problem, what is important is the basic foundation of the functionality of a heat pump system which the project will have as an implementation case.

Wi-Fi HaLow based on the 802.11ah is a relatively new protocol that is to be integrated and officially used in the market around the world. Although its specifications have been released years ago its still in an early phase of development and testing, and not especially prevalent in the market yet.
Based on the growing market for IoT, the new 802.11ah (Wi-Fi HaLow) was developed to deal with requirements for IoT. The specification for Wi-Fi HaLow is said to offer the best combination of operative performance in form of among other things as range, power consumption, and throughput. The protocol also has several interesting collaboration methods and features towards existing networks, Target Wake Time (TWT), Access Window (RAW), and native IP support. Overall to present Wi-Fi HaLow‘s specification and compare its performance towards other LPWAN protocols and perform tests that take into account realistic variables is interesting and necessary to understand the protocol operative functions.

1.3.3 Hardware and Software Selection

The project consists of different microcontrollers that handle input and output. In a smart home environment where sensors collaborate it is often only required to have a light-weighted microcontroller to handle the signals, but in this project, it shall also be developed an own Graphical User Interface(GUI) for the home center, so some computation is needed. As for hardware the two versions of Raspberry Pi‘s are chosen, Raspberry Pi 3 for the home center that handles the GUI and a Raspberry Pi Zero as the zone controller. As the programming language, Python has been chosen cause of its ease-to-use and most of the online guides towards the use of Raspberry Pi.
The project has front-end development which is consumer-related through QoE (Quality-of-Experience).

1.4 Report Outline

The remainder of this thesis is organized as follows:

Chapter 2:
The second covers the theoretical basis for understanding the project topics which covers issues regarding the project objectives and research questions. This also includes the theory for developing the test application used in further chapters.
Chapter 3:
The third chapter focus on the design and implementation of the components to fulfill testing in the chosen environments
Chapter 4:
The fourth chapter covers test results obtained during chapter three, this covers results of performance and deficient test objectives.
Chapter 5:
The fifth chapter covers a discussion of the obtained results presented in Chapter 4 and elaborates parts that could have been better. The chapter also draws an overall conclusion for the project.

Chapter 2

Related Work and Enabling Technologies

This chapter provides theoretical related work and challenges towards IoT and the affected structure towards this thesis.

2.1 Smart Home: What and Why?

This section provides the general view of the smart concept and communication protocols.

2.1.1 The Smart Concept

The smart term in smart homes, smart grids, or smart industry is the process to gather information and automate functions to improve sections such as cost savings, increased Quality of Life(QoL), security/safety, or surveillance purposes such as protect critical systems.
In general, a smart home system consists of two components: Sensors that measure relevant conditions in the home environment, while the smart home hub lets you connect all your smart home sensors together to create automatic functions. In the smart home, the collaborations between the sensors are complex that need to be understood. A research paper from the Multidisciplinary Digital Publishing Institute presents the compatibility problem in the smart homes sector, and further discussing the complexity that consumers of smart home technologies have as a repercussion of the many protocols available [12].

Smart home central is also referred to as "Smart hub". A smart hub is the main unit of the smart home, and its main purpose is to manage the data that is being exchanged between the components of the smart house [13].

The GUI that is often implemented in the smart hub is for the user interaction with the self-automated system is to configure preferable preferences, such as temperature, security measurements, or other environmental factors. The smart home hub act as a brain that collects environment data to take actions based on pre-configured parameters by the company that made the hub or configure based on the preference of the owner/user of the system.
All the preferred devices should be compatible with the central unit. If the central can manage the different protocols that devices of different providers use to communicate, the smart home system would be more dynamic and easy to integrate new devices. The components of a smart home system transfer and handles information with the help of protocols that are optimized for low-powered components, such as phones, microcontrollers, or other battery-driven devices.

The smart home's main feature is to make things automated and be controlled based on the user's preference; this also makes opportunities for cost-savings and convenience [14]. Controlling the environment and components through the phone can make the house's many chores and operations simple for the users. For example, set the coffee machine to make coffee while you are in bed in the morning or heat the living room a couple of hours before you wake up can be comfortable for many.

It can also help with more critical systems such as managing data regarding surveillance of the houses many sectors or help disabled with their many challenging tasks [?]. Further in this thesis, it will be presented a system where smart home automatic functions and surveillance perks can be used to deal with problems that can occur.

This also refers to the IoT concept which will be explained in chapter 2.1.2.

2.1.2 IoT; Functions and Considerations

IoT is a structure based on connecting electronic devices, sensors, and other devices to the Internet. IoT is a complex system that is made for data gathering used for surveillance and decision making with or without human interaction, which further can make operations automatic to improve the QoL (Quality of Life). IoT is used in applications such as smart grid, E-health, agriculture, or smart homes to improve operational factors such as efficiency, wellness, health, or conservation.
This complex evolved from M2M (Machine-to-Machine) communication, it is no longer just machine-to-machine anymore, now the M2M is currently being exchanged by the MTC (Machine Type Communication) to better describe the complexity of the system network

collaboration [1]. Further connecting these MTC systems to the Internet engaging the new network communication type IoT.

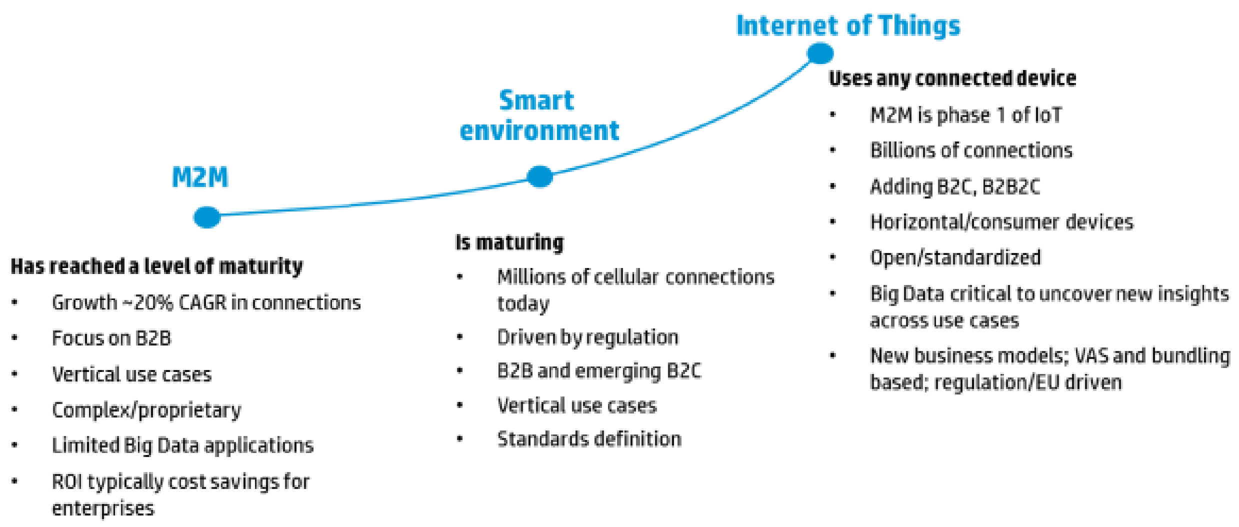


Figure 2.1: Evolution of M2M to IoT [1]

Nikola Tesla, which is a great engineer, said in an interview in 1926:
"When wireless is perfectly applied the whole earth will be converted into a huge brain, which it is, all things being particles of a real and rhythmic whole. We shall be able to communicate with one another instantly, irrespective of distance. Not only this, but through television and telephony we shall see and hear one another as perfectly as though we were face to face, despite intervening distances of thousands of miles; and the instruments through which we shall be able to do his will be amazingly simple compared with our present telephone. A man will be able to carry one in his vest pocket" [15].

Through the development and advancement in digital signal processing, sensors, communication, and artificial intelligence, this "brain" that Nikola Tesla spoke of have more or less been applied to the earth communication methods [16].
Now the IoT concept is applied in video streaming, smart environment (such as energy grids or/and smart homes) and E-health [17].
IoT will not only provide benefits for private users to increase QoL, but it will also bring market opportunities for equipment manufacturers, ISP (Internet Service Providers), and application developers. A study from 2015 estimates the projected market share of dominant IoT applications by 2025, see figure 2.2 [2].

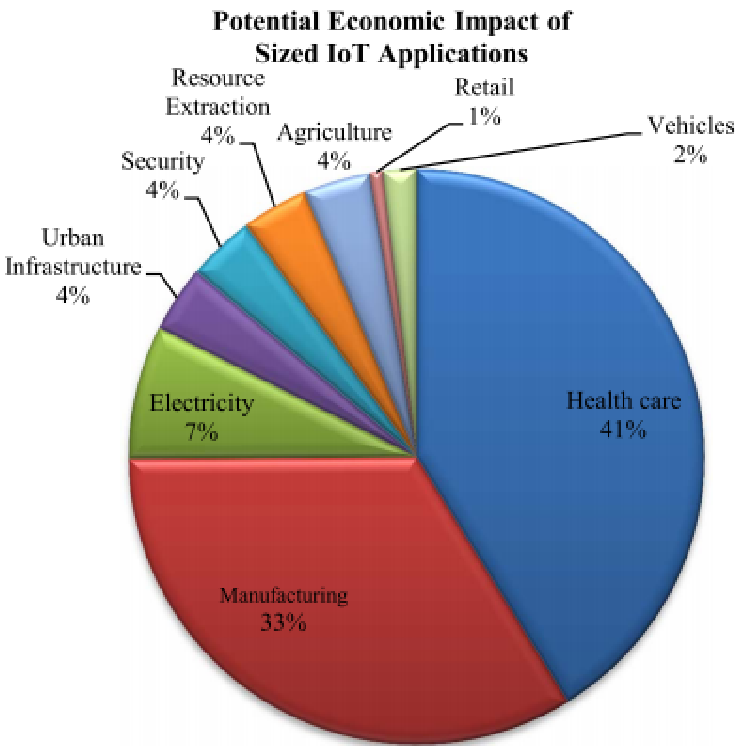


Figure 2.2: Market share of IoT applications [2]

This project will focus more on what lays behind the system's operational scope for the users of a smart home system, and what possible operations a smart home system can be used for with or without IoT.

2.1.3 Protocol Overview

Most IoT devices have limited resources in terms of power, processing capabilities, bandwidth, etc. Choosing the right protocol is critical to IoT devices. Protocols have simply defined the format and rules of how packages should be sent through a system. Different protocols have also unique scope and limitations of systems they are built for, some are built for high speed and others are built for low speed and more reliable transmission. Choosing the right protocol depends on the application, for example; can the application tolerate a high packet loss or high latency?

Figure 2.3 shows some of the most prominent protocols on the market to this date.

Layer		Protocols
Application Protocol		DDS, CoAP, AMQP, MQTT, MQTT-SN, XMPP, HTTP REST
Service Discovery		mDNS, DNS-SD
Infrastructure Protocols	Routing Protocol	RPL
	Network Layer	6LoWPAN, IPv4/IPv6
	Link Layer	IEEE 802.15.4
	Physical/ Device Layer	LTE-A, EPCglobal, IEEE 802.15.4, Z-Wave
Influential Protocols		IEEE 1888.3, IPSec, IEEE 1905.1

Figure 2.3: Most prominent protocols [3]

2.2 Home Automation System

This Section presents an overview of the thesis's first objective towards design and implementation for a thermal supply system.

2.2.1 Why a Sensor System?

The collaboration between flow and temperature sensors is to prevent adverse events that could damage the thermal system's internal equipment. Lack of flow and inaccurate temperatures can cause the dynamic fluid repercussion to damage the heat pump and cause an ineffectual process.

2.2.2 Issues; Overview and Comparability

Smart home technology has grown exponentially in the last few years, but it isn't close to its full potential. The technology continues to develop and improve with the upcoming years, but one of the main concerns to this date lays with comparability and collaboration in the market. From a user perspective and non-technician people it is also a problem with cost and complexity [18].

2.3 MQTT

MQTT (Message Queue Telemetry Transport) is a lightweight protocol that is easy to use and implement and is a well-known communication protocol used in many applications. The functionality of this protocol fulfills the requirements for the project data collection system presented in section 3.2.1 and is therefore chosen as the communication protocol.

2.3.1 Intro and Functionality

MQTT uses a publish/subscribe model and is a band-efficient message transport protocol and is widely used in Internet of Things (IoT) systems, or other constrained environments. Because of its lightweight performance, it is used widely in different industry sectors such as automotive, logistics, manufacturing, smart homes, oil&gas, and transportation. It has then become an OASIS standard for communication within IoT applications [19]. The MQTT achieves a simple implementation with easily configurable features such as Quality of Service, Retained Messages, Persistent Session, Last Will, and Testament, Keep-Alive. The topology is a composition of a broker(s) and client(s) that subscribe or publish messages The normal composition is one broker and multiple clients as shown in figure 2.4.

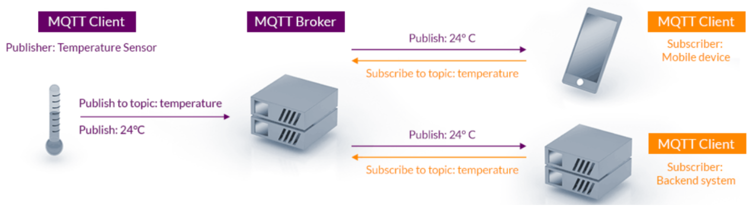


Figure 2.4: MQTT topology [4]

The broker is the main entity of handling messages, it uses subject-based filtering of messages by using topics. Each message contains a topic that the broker uses to determine which client should further receive this message. A client receives this message if the MQTT uses subject-based filtering of messages. Every message contains a topic (subject) that the broker can use to determine whether a subscribing client gets the message or not.

2.3.2 Topics

A topic is in the format of a hierarchical namespace, e.g., "Home/floor2/Office/Temp", and is used to filter the message from the different clients and sensors. There are also content-based filtering and type-based filtering that are available; even tho topic/subject filtering is the most common one to being used [20].

Regarding topics, there are also features for topic subscription *wildcards*, by using this in the topic format, clients can subscribe to more than one topic at a time by subscribing to a higher level. This is by using a plus sign "+" instead of the level the user wants subscribing to. An example is when subscribing to a higher level: "Home/floor1/+/Temp", by subscribing to a multilevel of topics, the plus sign "+" needs to be replaced by a hash-symbol "#" "Home/# "This lets the user receive all sense data below "Home".There is also a special topic feature that is not standardized yet; this is by using the "$"11 sign at the highest level of the topic. This can be used by the user to receive data from the broker for internal statistics and is usually in the format of "$SYS/," these statistics can be such as up-time, total message sent, or a list of clients that are connected to the broker [21].

2.3.3 Packet Exchange

During the set-up of a connection between the client and the broker in the MQTT topology, the entities exchange information of their operative state and configuration of how they shall exchange and handle the messages and when certain events are happening. All functions and configuration is exchanged at the start of the MQTT connection establishment with the CONNECT message [5].
Figure 2.5 shows the packet exchange between the client and the broker during the TCP connection establishment and the MQTT message exchange.

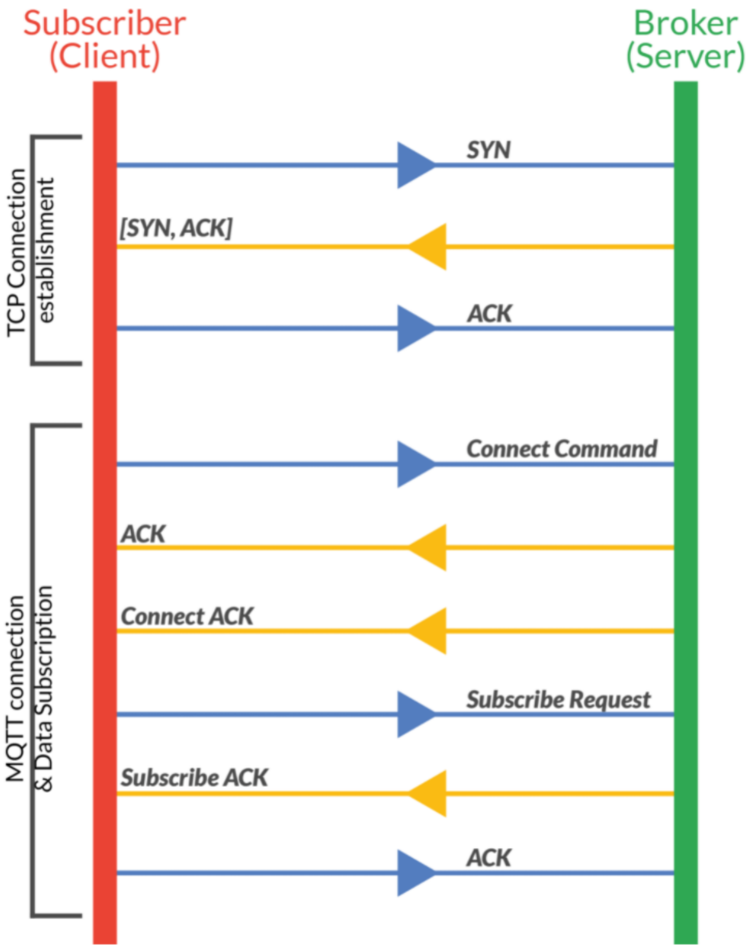


Figure 2.5: MQTT packet exchange [5]

Figure 2.6 shows the CONNECT packet format containing necessary information that is exchanged between the client and the broker.

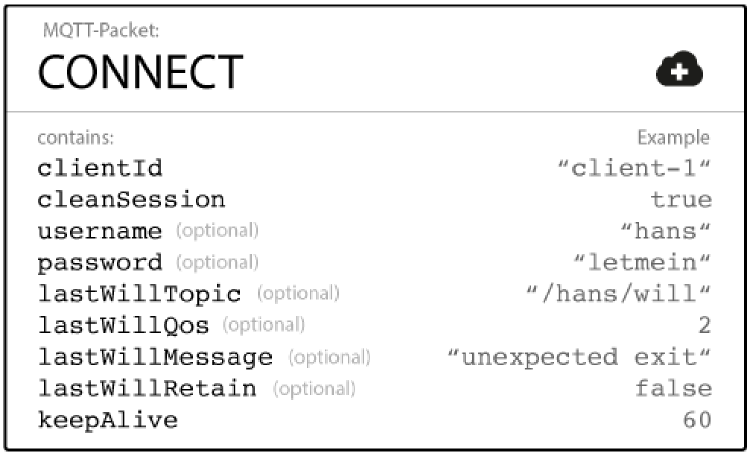


Figure 2.6: MQTT packet format [6]

2.3.4 MQTT Features

Below is described most of the relevant features that MQTT have considering the thesis need for existing and further development.

QoS Levels

QoS is the mechanism in communication networks to arrange the priority of different packets to ensure the most critical packets get the higher assurance to get transported faster and more reliable to their destination. Higher QoS labeled packets are have more statistically lower packet loss, latency, and experience a lower jitter effect when being transported [22].

MQTT arrange QoS priority into three levels [23]:

- At most once (0), this service level guarantees a best-effort delivery. Fast, but unreliable.
- At least once (1), this service guarantees that a message is delivered at least one time to the receiver. well-balanced option considering speed and reliability.
- Exactly once (2), this level guarantees that each message is received only once by the intended recipients, the slowest, but safest option.

MQTT uses different acknowledgment to guarantee the packet have arrived at the destination, but as a consequence, it uses more time as there are more packets exchanged between the client and the broker [23].

Retained Messages

Retained messages in MQTT are messages that contain info about the system and should be used when the system wants to inform newly connected clients about the existing system status and operative state. The retained messages are connected to topics, which means if a client connects to an MQTT network where a broker has a retained message on that specific topic that the client subscribed to, the broker will share that associated message [24].
The Retained message in MQTT is set by configuring the retained flag in a publish message to either true or false based on preferred functionality of the function [24].

Keep Alive

Keep Alive is a mechanism to deal with the *half-open connection* in the TCP functionality in MQTT. The half open-connection is an event where the (SYN/ACK) and ACK exchange between the client and server is not synchronous. Causing the server to get overflown by "pending messages", this can also be seen in different TCP attacks [25].

This caused massive packet loss because the one part that is operational in the TCP connection will progress normally to only overflow the other part that is handling the queue operation.
The keep-alive mechanism in MQTT uses two packages to overcome this problem. The first package is a *PINGREQ* sent from the client to the server to notify the server that is still "alive". The other packet is where the server receives the PINGREQ from the client and responds with a *PINGRESP* to the client to notify that the server is still available [26].

Persistent Session

Persistent Session in MQTT is where the broker stores some information exchanged when a client establishes a connection. This is to overcome additional work if the client gets disconnected from the broker. Where am event where an interruption is happening causing the client disconnect from the broker in a non-persistent session, information about the connection and topics are lost. This will mean that the client has to re-establish the connection every time it disconnects, and so exchanging information again with the broker. If the client and broker establish a persistent connection much of the information will be stored on the broker [27].

Last Will and Testament

Last Will and Testament (LWT) is a message that is being stored after the client and the broker establish a connection. If the client with LWT stored at a broker has an "ungracefully" disconnect from the system, the broker will notify the content of the LWT to other clients that are connected to the same topic. If the client with an established LWT disconnects normal, the LWT will be discarded [28].

2.3.5 Security

The project doesn't focus on security within communication protocols, but to understand the functionality and usage of MQTT it is necessary to dig briefly into its structure of security measurements.
MQTT security features are divided into 3 layers of the OSI model; Network layer, Transport level, and application-level [29]. At the Network layer security measurements, it is recommended to use a physically secured network or implement a VPN (Virtual Private Network) to secure the MQTT application. At the Transport layer, MQTT commonly uses other state-of-the-art security standards such as TLS/SSL to maintain confidentiality within the system. This means that data can't be read during transmission and provides client-

certificate authentication. Further, the security at the application layer provides configuration and implementation corresponding to the transporter layer authentication methods (Username/password) and possible encryption methods [29].
As the security measurements prevent attacks there is a trade-off between the degree of usability and security. The system described in chapter 3 will be using standard configuration as the system is non too little risk of attacks.

MQTT Client and Server implementations SHOULD offer Authentication, Authorization, and secure 358 communication options, such as those discussed in Chapter 5. Applications concerned with critical 359 infrastructure, personally identifiable information or other personal or sensitive information are strongly 360 advised using these security capabilities

2.4 Wi-Fi HaLow: 802.11ah

This section provides an overview of the 802.11ah based protocol called Wi-Fi HaLow. As this thesis is to test the range and performance of the protocol the section has a focus on factors that could affect the outcome of the testing procedure.

2.4.1 Overview

Wi-Fi HaLow which is based on 802.11ah standard is an open-source that operates in the sub 1GHz spectrum, which offers long-range and low power connectivity. This enables varieties in new power-sensitive IoT use-cases such as smart homes, vehicular communication systems, industry, and healthcare [30].
The main benefits of this protocol is the frequency spectrum used, new Target Wake Time (TWT), Restricted Access Window (RAW) [31] [32], extended max idle periods, and the absent need for proprietary hubs or gateways [30].

2.4.2 Scope and Purpose

The scope and purpose for Wi-Fi HaLow are targeting IoT environments where the devices require long-range connections where the path often has obstacles, these devices also need to operate on a single battery for a longer period (month/years). Wi-Fi HaLow tends to solve these problems by implementing new features to fulfill the requirements for enhancing existing and new IoT applications.

2.4.3 Frequency Band and Range

The IEEE 802.11 has defined a new protocol called 802.11ah, which operates in the sub 1 GHz license-exempt band [30]. Operating in an unlicensed band could bring complications and interference issues to other devices that use the same spectrum. So why use this spectrum?
All radio devices are divided into the electromagnetic spectrum (EMS), where the different devices operate depends on the operating function of the devices, such as how much data should be transferred or what range restriction does the equipment needs to operate in. Increasing the wavelength means higher data rate, but decreases in energy, which results in poor penetrating through walls [33] [7].

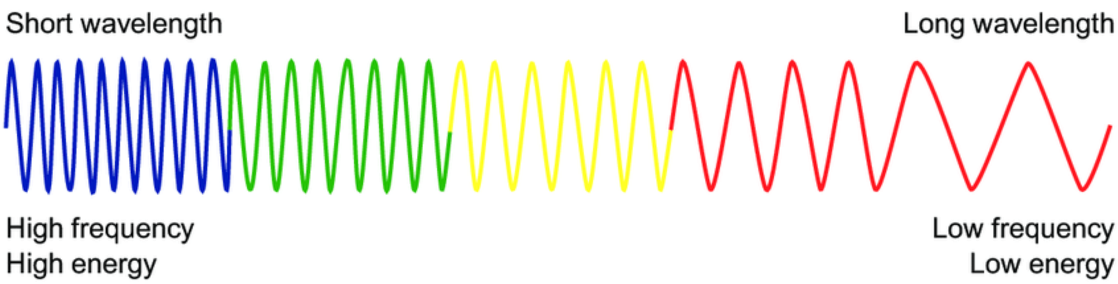


Figure 2.7: Energy vs wavelength [7]

So in short terms, which frequency to choose is to look at the operating function of the application. Further, this brings for a new question regarding the EMS, why does 802.11ah operate in the unlicensed band? Many private organizations work in the unlicensed band, mostly because of permissions and in terms of money. Such as cellular networks work in licensed bands to avoid interference with other radio equipment. Cellular networks mainly operates in low frequencies to compensate for the long transmission distance [34] [33].

Wi-Fi HaLow falls under the category of Low Power Wide Area Network (LPWAN) which is categorized for communication range up to several kilometers [9]. 802.11ah is supported in several countries but differs in what spectrum that the protocol is supported in. The supported bandwidth is regulated by the specific country, Figure 2.8 shows the global channelization of 802.11ah [34].

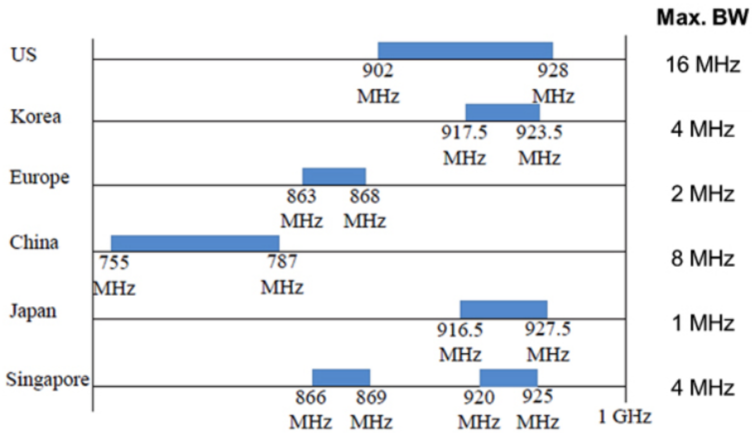


Figure 2.8: Wi-Fi HaLow frequency band [8]

Utilizing the frequency benefits of the protocol with configuration in the physical layer is used to optimizing the overall performance to achieve a higher bitrate at a longer range.
The physical layer is the proposition of optimizing the factors of data rate and range, which Wi-Fi Ha-Low is fulfilling performance of a transmission range up to around 1,2km depends on the modulation scheme used. To achieve the preference performance to a given application technical requirement, an MCS (Modulation Coding Scheme) index table can be used. The MCS is a composition of different metrics to index the theoretical performance between two Wi-Fi stations. There are currently 5 parameters that are to be known to acquire the metric index, but it also relies on SNR (Signal To Noise Ratio) and RSSI (Received Signal Strength Indicator) [35].
MSC metric factor composition [35]:

- Modulation Type; Configure phase and amplitude modulation to a higher type (e.g 64-QAM to 256-QAM) results in higher possible information transferring, but makes the signal more vulnerable to noise.
- Coding Rate; The coding rate is the contrast between how many bits are being used for data transferring and error correction.
- Spatial Streams; This factor dictates how many streams are used to transfer data, a higher value increases the data rate, but also increase the vulnerability for noise and interference.
- Channel Width; The width of the channel has a major impact on how much information can be transferred, but an increase in channel width will result in a lower SNR. Doubling the number of channels will increase the noise floor by 3 DB.
- Guard Interval; A guard interval is simply an interval between each transmission. Decreasing the interval will result in more transmission in a shorter time interval, but

will also increase the interference.

So overall, there is a compromise between SNR and data rate, which is mostly dependent on the system requirement and usage.

2.4.4 LPWAN Comparison

802.11ah has a hybrid composition of range and data rate insights, which makes it suitable for the intended applications concerning IoT communication networks.
The main concept of LPWAN technologies is to address problems like long distances and low power consumption. The main traits and similarities between common technologies are shown in Figure 2.9. Each of the technologies claims different ways to solve IoT challenges.

	Wi-Fi HaLow™	LoRaWAN	Sigfox	NB-IoT
Modulation	OFDM	CSS	BPSK	QPSK
Frequency	Unlicensed ISM bands (863-868 MHz in Europe, 902-928 MHz in North America)	Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia)	Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia)	Licensed LTE frequency bands
Bandwidth	2/4/8 MHz	250 kHz and 125 kHz	100 Hz	200 kHz
Maximum data rate	150 kbps to 78 Mbps	50 kbps	100 bps	200 kbps
Bidirectional	Yes / Half-duplex	Yes / Half-duplex	Limited / Half-duplex	Yes / Half-duplex
Maximum messages/day	Unlimited	Unlimited	140 (UL), 4 (DL)	Unlimited
Maximum payload length	1500 bytes	243 bytes	12 bytes (UL), 8 bytes (DL)	1600 bytes
Range	1.5 km (urban)	5 km (urban), 20 km (rural)	10 km (urban), 40 km (rural)	1 km (urban), 10 km (rural)

Figure 2.9: Technical specification comparison [9]

Chapter 3

System Design and Implementation

This chapter presents the design and implantation of the objectives. Chapter 3.1 is presenting an overview for Chapter 3.2 and 3.3 for an automated home system with a connection to a thermal water supply system.

Further, Chapter 3.4 presents design and implementation towards an evaluation procedure for testing 802.11ah with restrictions to Norwegian law and regulations.

3.1 System Design and Overview

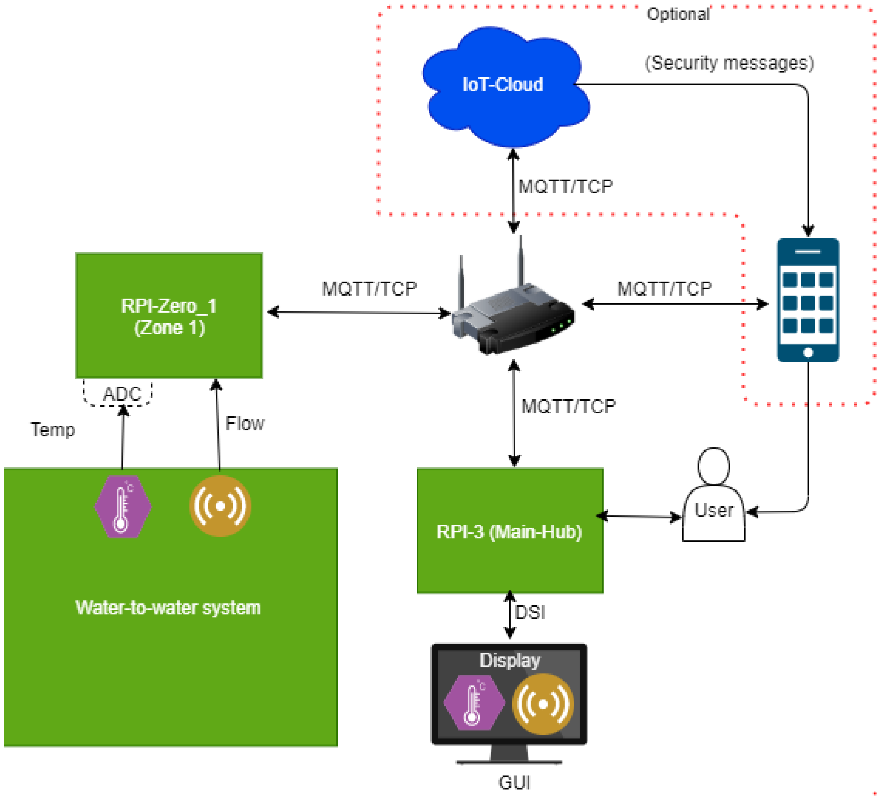


Figure 3.1: Overview of Data Collection system

Overview

The system shall consist of two Raspberry Pi's and two add-on cards that are in charge of processing the information from sensors or/and visualize data to the user. The user shall then have the opportunity to read the data from the RPi-3's 7-inch display. The sensors shall be considered in a real-life application that could be relevant to sectors of a smart home. For the enhancement procedure, the RPi-3 connected display should visualize the data through a user-friendly GUI that should be capable of adding new sensors to the system and visualize the data through a systematic perspective.

Main HUB and Zone Controller

The chosen entities shown as *RPI_Zero_1* and *RPI-3(Main-Hub)* in Figure 3.1 that manage signal conversion and data representation is two Raspberry Pi's. These units will work as back-end and front-end devices.
The reason for choosing Raspberry Pi's is because of theirs high availability of addon cards that makes them suitable for many different systems. They have a high custom-able interface

that mainly runs on a Linux-based operative system(OS), which makes python programming a good choice for developing programs.

GUI

The display shown in Figure 3.1 is an 800 x 480 display that connects via an adapter board that handles power and signal conversion. The conversion converts the parallel signals from the display to the serial (DSI) port on the Raspberry Pi.
Further, the power that is recommended to power the RPi-3 alone is 5 volt and 2A; therefore, the adapter card has an additional power input for powering the additional setup. This is not required but highly recommended due to negating any voltage drop caused by an excessive current draw from the GPU or/and the CPU to heavy load [**?**].

MQTT/TCP

MQTT/TCP will be the main data communication protocol between the devices.

Optional: IoT Cloud and Phone Interface

Implementing IoT-Cloud and a phone interface in the project is planned, but not implemented in this project. Due to the broad specter in this project, it was chosen to present the design for the solution, further to present pros and cons the implementation can cause but exclude its practical execution.

3.2 Data Collection; Thermal Water Supply System

This section covers the thermal system overview and sensors used for data collecting.

3.2.1 System Overview

An illustration of the water-to-water system is presented below in Figure 3.2.

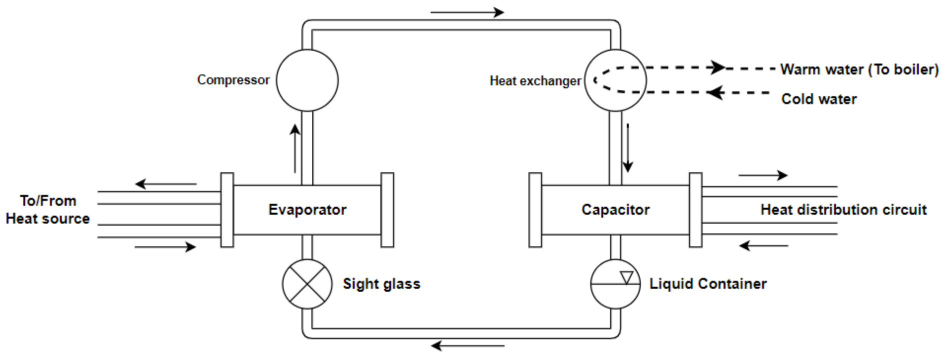


Figure 3.2: Water-to-water system

3.2.2 Zone: Thermal System

To solve these problems that can occur in the thermal system, the project implements four temp sensors and two flow meters to monitor the system, see figure 3.3.

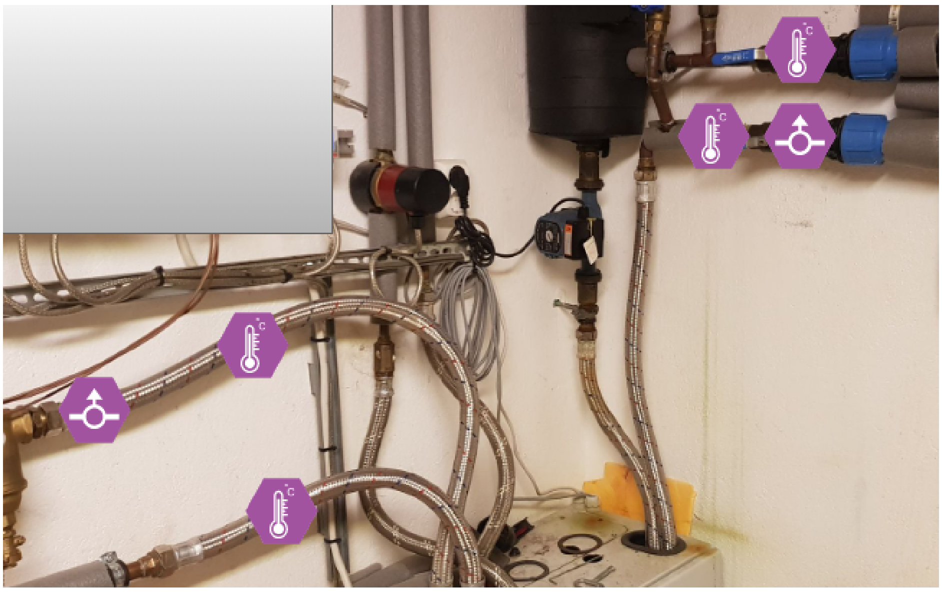


Figure 3.3: Water-to-water system with sensors

With six sensors strategically placed to measure flow and temperature, surveillance of the system can avoid circumstances explained in Subsection 2.2.1. See Figure 3.4 for illustration.

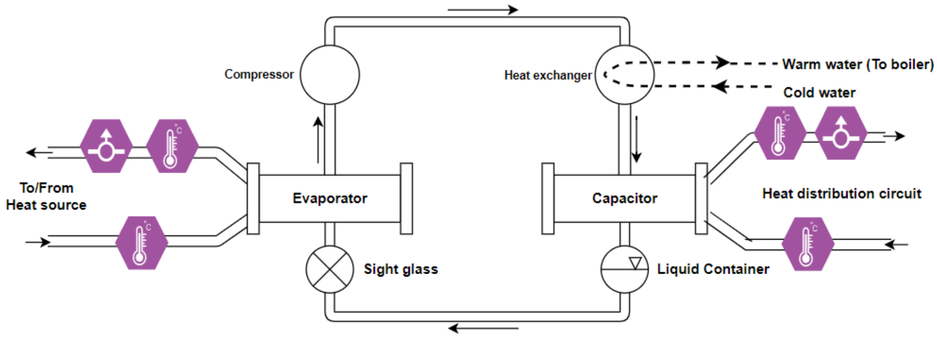
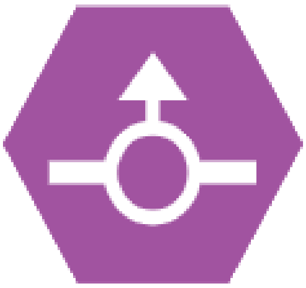


Figure 3.4: Illustration of the thermal system with sensors

Flow Meter

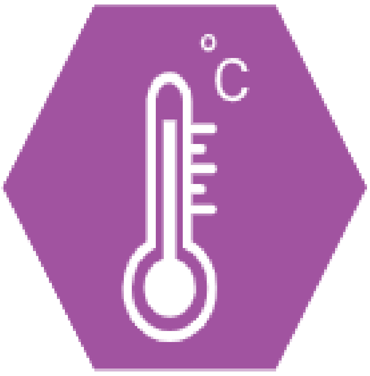


(a) Flowmeter symbol

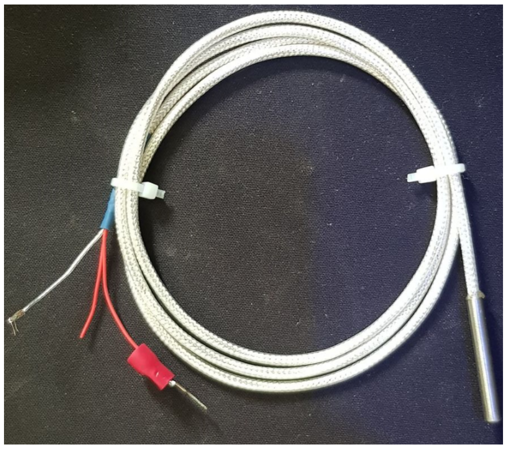
(b) Flow meter

The symbol presented in Figure 3.5a represents a DIGITEN G1-1/2" G1.5" hall effect sensor 3.5b used in this case to measure the water flow in pipes. It has an operating range from 5-150L/min and producing an output square wave pulse signal.
Calculating how much water is going through the flow meter is a formula that composite of multiple parameters that take into account pipe dimensions and pulses generated by the hall effect sensor. A simplified formula is provided by DIGITEN company with a K factor of 0,5 : $0.5 \times 60 = 30$ pulses/L.

Temperature Sensor



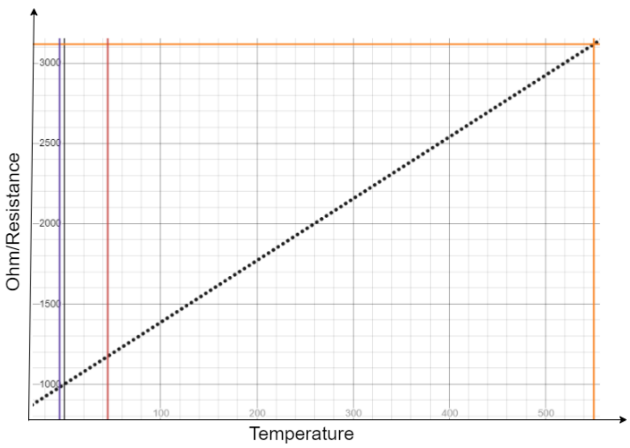
(a) Temperature sensor symbol



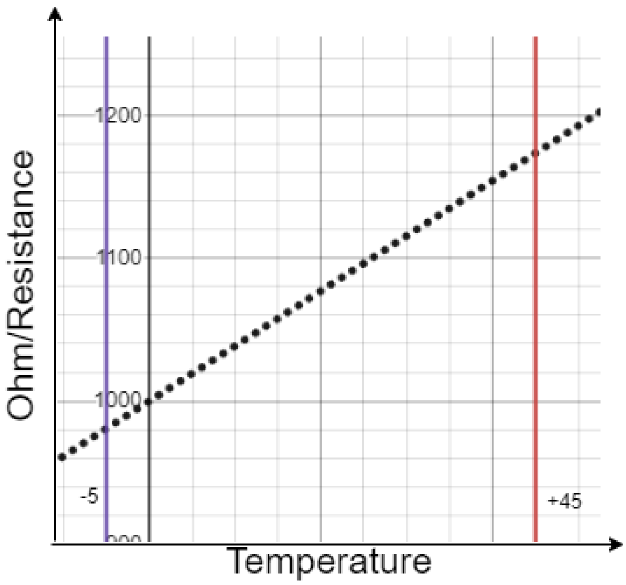
(b) Temperature sensor: PT-1000

The symbol presented in Figure 3.6a is representing a temperature sensor, this sensor is a PT-1000 element shown in Figure 3.6b. The PT-1000 element is a Resistance Temperature Detector (RTD), which means it is a resistor that changes resistance in ratio with temperature and has 1000Ω at 0°C . It has very ease-of-use which only requires measuring the resistance between 2 wires. For optimizing the precision of the measurement there is an option for a 3-wire (and 4) solution, this includes subtracting the wire resistance from the equation, as a 1°C change will result in a 0.384Ω change in resistance [36].

The specification area of the PT-1000 element is up to 550°C sensor is shown in Figure 3.7a, while the project typically operative area is shown in Figure 3.7b:



(a) The chosen PT-1000 Operative area, up to 550°C



(b) Operative area of the thermal system

Implementing a resistance measurement function requires an ADC (Analog-to-Digital Converter) as the resistance measurements are analog and the raspberry pi doesn't have an integrated ADC. A data acquisition card for the raspberry pi is needed for handling the analog signals. For this project, it is chosen the Mega-RTD eight-channel card from Sequent Microsystems. This card communicates with the raspberry through the I^2C(Inter-Integrated Circuit) bus that only requires 2 pins [37].
The formula for converting analog signal in Ohm to temperature is given by:

$$R(T) = R0[1 + \alpha(T - T0)$$

where:
R(T) = Resistance measured at temperature T
R0 = Resistance where temperature is zero degrees
α = Coefficient of resistance change factor in relation with temperature
T = Temperature in degrees
T0 = The reference temperature for R0

Calculating the relationship between temperature and resistance is based on the IEC 60751:2008 standard. The IEC 60751:2008 standard specifies the requirements and temperature/resistance relationship for industrial platinum resistance temperature sensors [38].
Converting this formula concerning T using the specification for the PT-1000 element, the converting function is given by line 48 to 52 in Appendix C.

3.2.3 Graphical User Interface

Interaction design is the process of designing usable and user-centered interactive systems. What's important when designing a user interaction program is to consider factors such as ease-to-use and visual design of the information architecture. Both theory and practical techniques to support the development of usable interactive systems is important to consider. The concept of the development is to consider the context of the application and the user requirements. It is also important to have in mind the variety of required tools and methods for designing and prototyping the GUI. Including users during evaluation, design is crucial to cover the different aspects of the interactive system. The UI shall then utilize concepts from interaction design, visual design, and information architecture.

A simple method is to use the tk interface package that is available through python. Its structure method is build up in the format of a grid system with different geometric possibilities that could be defined to its intended usages, such as a button or just visualization of a parameter.

3.2.4 Hardware: Broker and Client Controller

The broker and the client uses the official raspberry pi Linux based OS.
The hub that is presented in Figure 3.8 operates as the broker and GUI is a Raspberry Pi 3b+ equipped with the official RPi 7 inch-display with additional add-on card to fulfill the requirement for power and signal processing to the screen.

Figure 3.8: Raspberry Pi 3b+ with add-on card and the official raspberry display

Further, Figure 3.9 presents the client with an add-on card that operates as an ADC and power supply for additional equipment e.g the flow meters and temperature sensors.

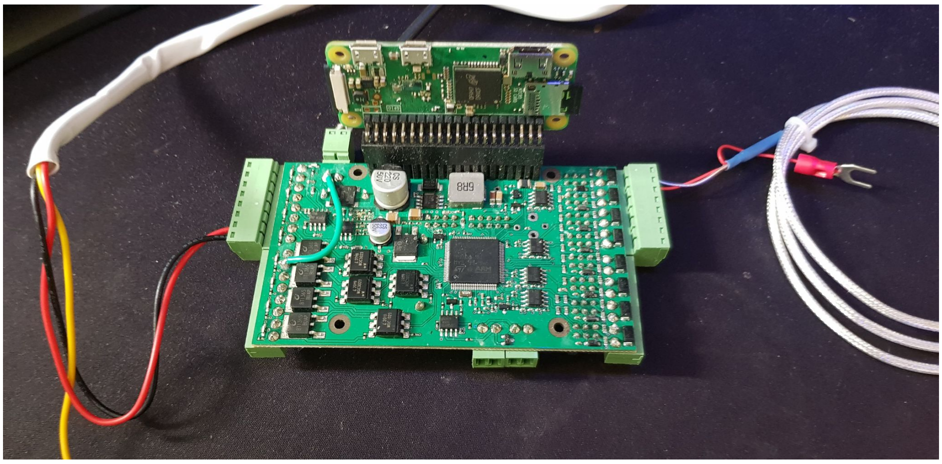


Figure 3.9: Raspberry Pi Zero with Megabas add-on card

3.2.5 Software Implementation

Implementing the software is done using python programming language and Thonny as the integrated development environment(IDE). Code shall follow a structured organization to make further development easier and easy to understand for external parties.
Code is compounded by several included libraries to support the add-on card operative functions and be able to construct an interface for visualization of the collected data.

3.3 Data Transmission: MQTT

The data transfer method is to be implemented and integrated into the data collection system designed and implemented in section 3.2.

3.3.1 Overview

Overview after integration:

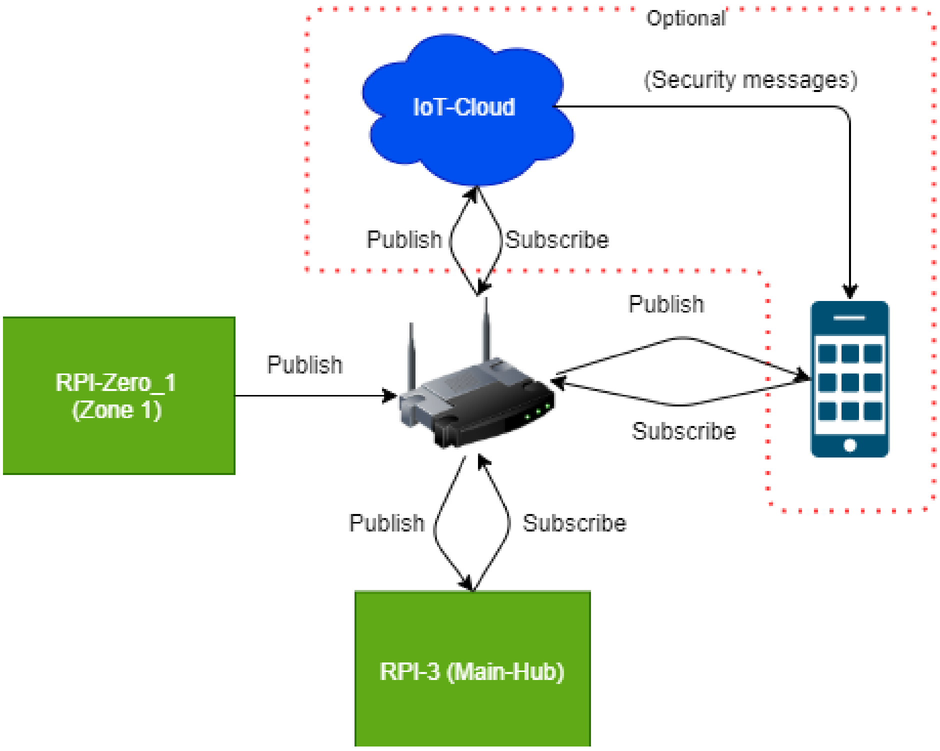


Figure 3.10: Overview of data transmission system based on MQTT

3.3.2 Features and Functionality

The functionality of the MQTT implementation is to exchange information between the devices in a defined interval that is convenient to the application.

3.3.3 Software Implementation

The RPi broker in the MQTT topology requires a *message broker*. A message broker is a middleware between the devices in the MQTT topology which translates the different messages from applications to the format of the MQTT topology [39]. The broker also manages the different messages like validation and routing, see Figure 2.2.Figure 2.2: Message broker handle messages. This software also works as a Message-Oriented Middleware (MOM), which enables applications with different languages to exchange messages [40].

In Appendix C line 9-11 and library in line 5 is all that need to implement the RPi Zero as a MQTT client, thus further the messages need to be published to the broker as presented in line 56 and 57. This function only requires two parameters that include the *topic* presented in Subsection 2.3.2 and the message itself, which is the sensor data collected.

3.3.4 Cloud

Connecting the MQTT architecture to a cloud makes it possible to access data regardless of user location as long it is within a range of signals that can provide internet access. Connecting the system to the cloud provides the system with new additional features which makes it more suitable for maybe more critical systems that require instant management.
Connecting the system to an external cloud provider, e.g., Google will let the user of the system access analytic methods and features regarding storage opportunities, additional features can be found here.

An available cloud solution is the Google Cloud IoT Core or HiveMQ Cloud. Google's IoT solution supports many of the preferable solutions that the system could need in different scenarios, such as QoS support, monitoring of data, and audit logging. This is a familiar system that utilizes to support more applications regarding a home environment with different needs and aspects. Cloud could be connected to systems that monitor operations that could afflict human well-being or other non-critical systems like automatic systems or surveillance systems.
Connecting the system to an external cloud provider, e.g., Google will let the user of the system access analytic methods and features regarding storage opportunities; additional features can be found here. One case is to implement an accelerometer in an application that can provide the system to sense vibration in, for example, pumps or fans, and then implement streaming analytics for real-time event-processing. If unstable vibrations are detected, the cloud provider can message the user's phone about the event that is happening, and the user can then manually check the system for faults before the system gets badly damaged. Implementing a cloud solution to the system will enable IoT functionalities through the

MQTT message exchange, which enables the user equipment to control and monitor systems regardless of location.

3.3.5 Testing MQTT

Testing the packet transmission between the devices is a necessary procedure to test if MQTT maintains its functionality. To test the network traffic Wireshark is used.
Wireshark is widely used as a tool for analyzing network traffic in real-time, it is a completely free program that captures network traffic in real-time on the local network. It is a well-known and suitable program for testing the MQTT communication protocol to analyze its functionality and mechanisms.
To install Wireshark on Linux is simply to use *sudo apt-get install wireshark* in the terminal. To monitor all packets transferred between the devices in the MQTT topology, it is necessary to install it on the broker (RPi 3B+). Wireshark has also filter mechanisms to let the program focus only on the necessary transmissions [41].

3.4 802.11ah

This section presents the implementation towards Wi-Fi HaLow. This procedure is executed by using evaluation kits from Silex Technology.

3.4.1 Overview

Figure 3.11 present the setup with two fully equipped modules using RPi.

Figure 3.11: Two evaluation kits used to test Wi-Fi HaLow

3.4.2 Testing

Testing the Wi-Fi HaLow in a realistic environment is crucial for an analysis process to the protocol. There are few testing possibilities except for simulations based on theoretical basics given the technical specifications of the IEEE group.
One possibility is the "SX-NEWAH Evaluation kit" for Silex Technology, which is a module powered by an NRC7292 SoC (System on a Chip) from Newracom. This is the first industry module that is used for testing the 802.11ah module that operates in the sub 1GHz band [42].

The module driver is based on Linux kernel version 4.14 which ports to the reference platform, which in this case it is used as an RPi 3b+. The driver collaborates with the platform to perform certain rigorous processes such as tests, evaluations, fixes, and enhancement to the driver that processes the 802.11ah protocol. This follows with Silex radio drivers to execute the test and analyzing tools for the evaluation process to test such as basic wireless functionality, security, data throughput, network modes (STA and AP / DHCP settings).

Test of the 802.11ah is performed by using Iperf which is a network tool used to configuring and measure network performance. Iperf configuration establishes a connection by setting up modules for client and server. The configuration The process then reports relevant test parameters such as bandwidth, loss, and other parameters.

According to SubSection 2.4.3 the frequency of 802.11ah in Norway (where the testing is to be performed) has been allocated the spectrum 863-868 MHz, while the SX-NEWAH product specifications testing equipment is developed to operate between 903.5 and 926.5 MHz. See SubSection 3.4.3 for regulations and measures to adapt to the testing requirement.

3.4.3 Restrictions: Law and Regulations

The use of frequencies and electronic devices in many cases requires permission from the national security authority, in Norway, it is called NKOM ("Nasjonal Sikkerhetsmyndighet"). NKOM is an executive supervisory and administrative authority for services within electronic communications in Norway [43]. NKOM manages the frequency resources in Norway. They provide information about free use, frequency licenses, which frequencies and applications are needed for TV and radio, Professional / Private Mobile Radio (PMR), hunting radio, amateur radio, and emergency beacon transmitters, and information about requirements for import and sale of equipment that uses frequencies.

To use the bought equipment 4 regulated conclusions must be followed:

1. **Principal When Using Electronic Devices in Norway**
   The main principle when using electronic devices in Norway is that the equipment must be CE-labeled. The project's main cause is to use the equipment for research and development of new technology and equipment.

2. **Frequency Band**
   The frequency range assigned is 915.8 - 919.4 MHz. In this band, the Norway defense also operates, which means it is important that the technical parameters are in accordance with the Norwegian frequency plan. More information about the use of this band, as well as reference documents, can be found on Nkom's frequency portal (National frequency plan - Frequency portal (nkom.no)) .

3. **Use of 802.11ah**
   The use of 802.11ah must be regulated by an ETSI standard ETSI TR 103 245, this standard reviews technical conditions for the use of 802.11 [44].

4. **Effect and Bandwidth**
   Both ETSI TR 103 245 and ERC recommendation (70-03) Rec7003e page 15 - Annex 3: Wideband Transmission Systems), Norway has implemented states that the permitted power from the equipment is a maximum of 250 mW ERP (Effective Radiated Power) and used bandwidth max 1 MHz.

With such conditions fulfilled, the equipment is subject to the free use regulations and does not need additional permission from Nkom to use it.

Solution and regulations towards Nkom regulations:

Point 1, CE-Labeling:
The CE-Marking is to inform the supervisory authorities that the basic safety, health, and environmental protection requirements are met for the equipment [45]. All Raspberry Pi

products have undergone extensive compliance testing and are available through the raspberry pi documentation on the official website [46].

Point 2, Frequency Band:
Adjusting the equipment after the requirement given by point 2 to fulfill the assigned frequency range at 915,8 - 919,4 MHz is done by *channel mapping*. The total frequency range for the SW-NEWAH module between 903,5 - 926,5MHz, with supported bandwidths of 1, 2, or 4 MHz. These frequencies are mapped to various channels in the 2,4GHz and 5GHz spectrum with protocol 802.11 A/B/G [47]. Each frequency/channel is mapped to a unique channel number.

Point 3, Use of 802.11ah:
The marked in wideband SRD's are expected and are currently growing rapidly, therefore it is an essential need for additional spectrum to accommodate the growth. The ETSI TR 103 245 presents requests modifications to the regulatory rules of the UHF 870 - 876 MHz and 915 - 921 MHz frequency bands to enable the operation of Wideband SRD's (Short Range Device) with advanced spectrum sharing capabilities in these bands [44].

Point 4, Radiated Power:
According to point **4**, the transmitting power has to be decreased from 1W to 0,25W and the bandwidth can max be 1 MHz wide. The SX-NEWAH Evaluation kit has configurable parameters to decrease the power and the bandwidth to the required values. The radiated efficiency is the "Ratio of power radiated by the antenna to the net power accepted by the antenna from the connected transmitter" [48].

3.4.4 Station and Access Mode

Following Stationary mode setup in chapter 9.2 and 9.3 in Silex Technology start-up guide.

The modules must be configured in each mode, one in station mode and the other in access mode to initiate the testing process. The guide progresses through a setup where transmission power, maximum aggregation size, and guard interval are basic parameters to set up a connection between the modules.
Configure the different modes are shown in Figure 3.12 for access mode and further Figure 3.13 for stationary mode.

①	sudo insmod /lib/modules/$(uname -r)/nrc.ko fw_name=nrc7292_cspi.bin hifspeed=16000000
②	sudo ifconfig wlan0 up
③	cli_app set txpwr 23
④	cli_app set maxagg 1 8
⑤	cli_app set gi short
⑥	sudo hostapd -B ~/sx-newah/conf/US/ap_halow_sae.conf
⑦	sudo ifconfig wlan0 192.168.200.1

Figure 3.12: Configuring the evaluation kit to access mode

①	sudo insmod /lib/modules/$(uname -r)/nrc.ko fw_name=nrc7292_cspi.bin hifspeed=16000000
②	sudo ifconfig wlan0 up
③	cli_app set txpwr 23s
④	cli_app set maxagg 1 8
⑤	cli_app set gi short
⑥	sudo wpa_supplicant -i wlan0 -D nl80211 -B -c ~/sx-newah/conf/US/sta_halow_sae.conf
⑦	sudo ifconfig wlan0 192.168.200.2

Figure 3.13: Configuring the evaluation kit to stationary mode

3.4.5 Configuration

Additional configurations had to take actions as the default settings exceeds the required values presented in 3.4.3. Changing parameters during the setup presented in Figure 3.12 or 3.13.

Transmission power is restricted to the regulations from NKOM presented in chapter 3.4.3 point 4. The transmission power parameter is used by **cli_app set txpwr** line following the power in dBm (desibel milliwatts).This parameter must be changed on both devices to maintain the regulation provided. 250mW is equal to 24dBm, equation:

$$dBm = 10 * \log_{10}(mW) = 10 * \log_{10}(250) = 24$$

Meaning the new command is **cli_app set txpwr 24**.

3.4.6 Range

Referring to Figure 3.14, the contemporary formula of Friis transmission equation a theoretical assumption of the outcome is:

$$PR = PT * \frac{Gt * Gr * \lambda^2}{(4\pi)^2 * d^n)}$$

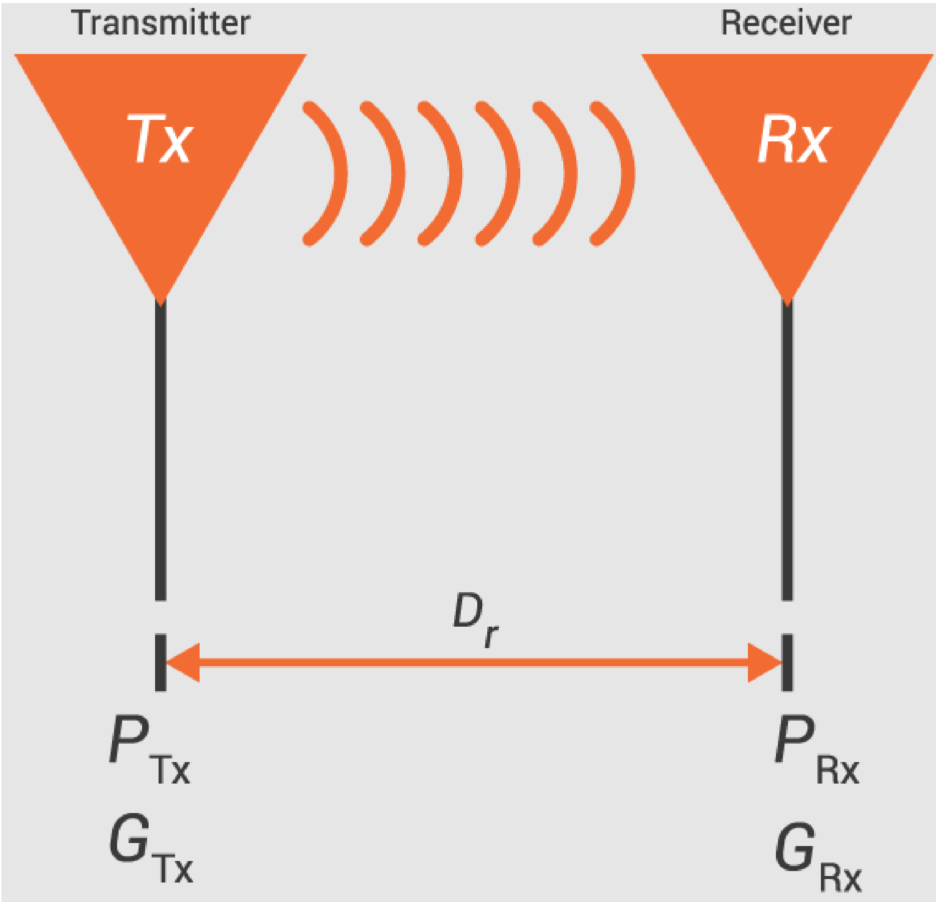


Figure 3.14: Friis antenna factors

where:

PR = Power available from receiving antenna

PT = Power supplied to the transmitting antenna

GR = Gain in receiving antenna

GT = Gain in transmitting antenna

λ = wavelength

d = Distance

c = speed of light

n = Spatial coefficient

where λ = c/f, c = speed of light, f = frequency
After a brief estimation of the range is should also be considered to optimize the system by using theorems that consider the *fresnel Zone*, see figure 3.15, considering its importance for the line of sight between two wireless systems to be free from any obstruction.

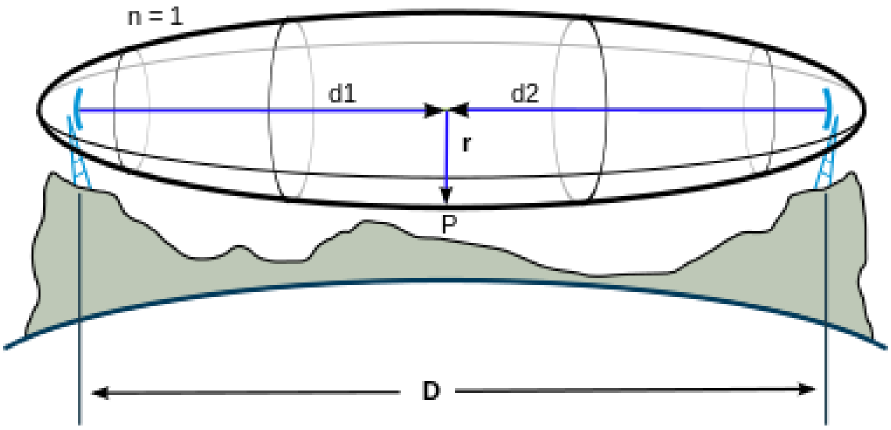


Figure 3.15: Fresnel zone

Constructing a theoretical elliptical region between the transmit antenna and the receiving antenna for optimizing the functionality and performance of the system [49].
Where the Fresnel Zone radius is given by:

$$R = \sqrt{\frac{n * d1 * d2 * f}{d1 + d2}}$$

where:
d = Link distance in km
n = Fresnel Zone number (should be greater than zero)
f = frequency of the transmitted signal

Considering these concepts gives a basis for testing and assumption considering range and performance further in the project.

Chapter 4

Testing

This chapter features tests and functional results of the objectives presented in Subsection 1.2.

4.1 Testing Scenarios

A given scenario for each of these thesis-developed systems is presented in this chapter to understand the physical environment and situation of the systems developed.

4.1.1 Scenario 1: Data Collection for Thermal Water Supply System

Testing Scenario for the thermal water supply system is comprehensive where each device and sensor is tested separately to acquire the required measurements and values. Later present the overall performance and functionality in a collected manner.

4.1.2 Scenario 2: Data Transmission Using MQTT

Testing the MQTT-based transmission between the devices using Wireshark is done after the devices in scenario 4.1.1 operate as normal. This scenario is where analyzing tool Wireshark is implemented on the broker/server to acquire the overall packet exchange between all devices in the topology presented in 3.3.

4.1.3 Scenario 3: 802.11ah

The last scenario is where the 802.11ah test phase and the acquired result are based on findings by the company Silex Technology. Testing of the 802.11ah could not be performed cause of a module error with the evaluation kit. During setup, there was an error with one of the evaluation kits causing the module that handles the 802.11ah functionality to be non-existing, see Subsection 4.4.1 Figure 4.8 for implementation error that occurred.
After several testing attempts, troubleshooting, and customer support with Silex Technology and Future Electronics, it came to an overall conclusion that it was and hardware error causing the trouble. Considering the time left with the project and the available resources it was decided to proceed further with existing test data from Silex technology.

4.2 Results and Testing: Data Collection for Thermal Water Supply System

4.2.1 Temperature Sensor

The temperature elements are tested separate from the ADC and the RPi zero to test if the program works as it should with converting the resistance value to temperature. Figure 4.1 is a direct measurement of the PT-1000 element resistance in a room temperature. Using the converting formula presented in Subsection 3.2.2 and Appendix C and using the acquired resistance of 1077Ωthe temperature is calculated to 19.9°C

Figure 4.1: Measured value of the PT-1000 element

Further, testing the code presented in C returns the following:

```
Python 3.7.3 (/usr/bin/python3)
>>> %Run Reg_MQTT.py

  Measured flow: 0
  Measured resistance: 1.089
  Measured Temp: 23.11
  RESET

>>> Measured flow: 0.6
  Measured resistance: 1.089
  Measured Temp: 23.11
  RESET
```

Figure 4.2: Measured values from ADC converted to temperature presented in degrees Celsius

The acquired values presented in 4.2 are measured under slightly warmer conditions but still operate normally.

4.2.2 Flow Measurements

A water rotor along with a hall effect sensor is present the sense and measure the water flow. When water flows through the valve it rotates the rotor. By this, the change can be observed in the speed of the motor. This change is calculated as output as a pulse signal by the hall effect sensor.
The flow meter is a hall effect sensor, which means that a given number of pulses represent a certain amount of water. When water flows through the flow sensor it rotates the rotor that generates a signal with help of the Hall effect sensor. This signal has been measured and presented in Figure 4.3.

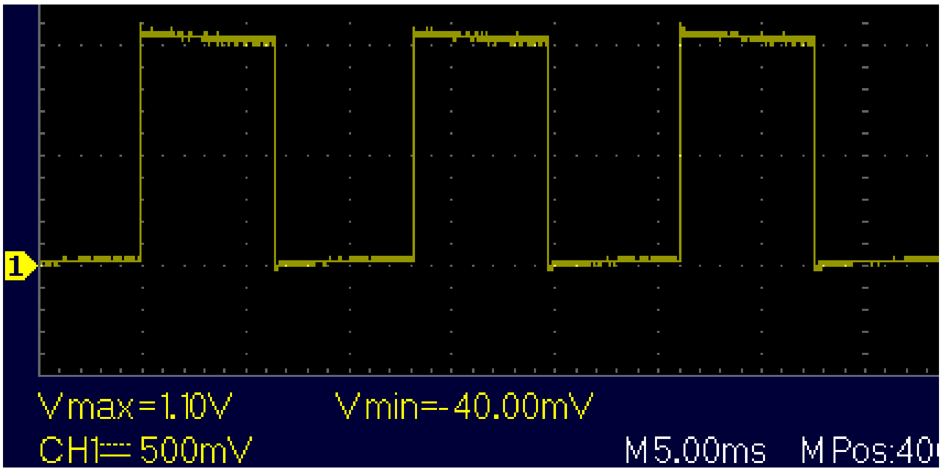


Figure 4.3: Measured signal from the flow meter in use

As the signal shows it's generating an expecting pulse signal which is proportional to the water flow. Given the formula presented in 3.2.2, more pulses over a shorter period mean more water flows through the flow meter.

The zone controller that handles input and from sensors is a Raspberry Pi Zero, this is the smallest version of the raspberry pi series and has smaller computation power compared to the others, but has more than enough computation power to handle the required signals.

4.2.3 GUI

The finished product of the GUI had high responsiveness with little delay switching between the different zones and screen layouts. As the GUI development was done on a windows computer using python there were some small converting issues, but require little to no effort to adapt to the raspberry pi.
Figure 4.4 shows the final layout of the home screen.

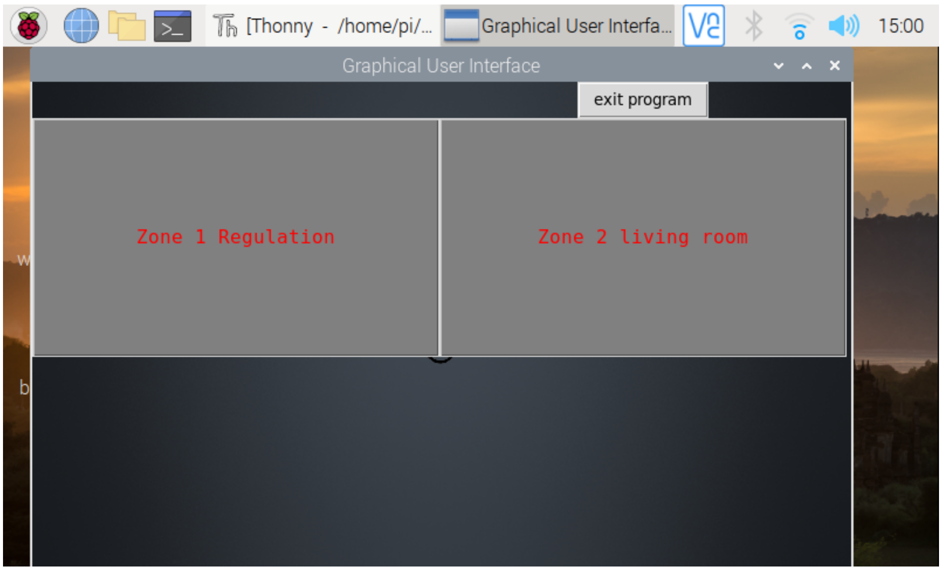
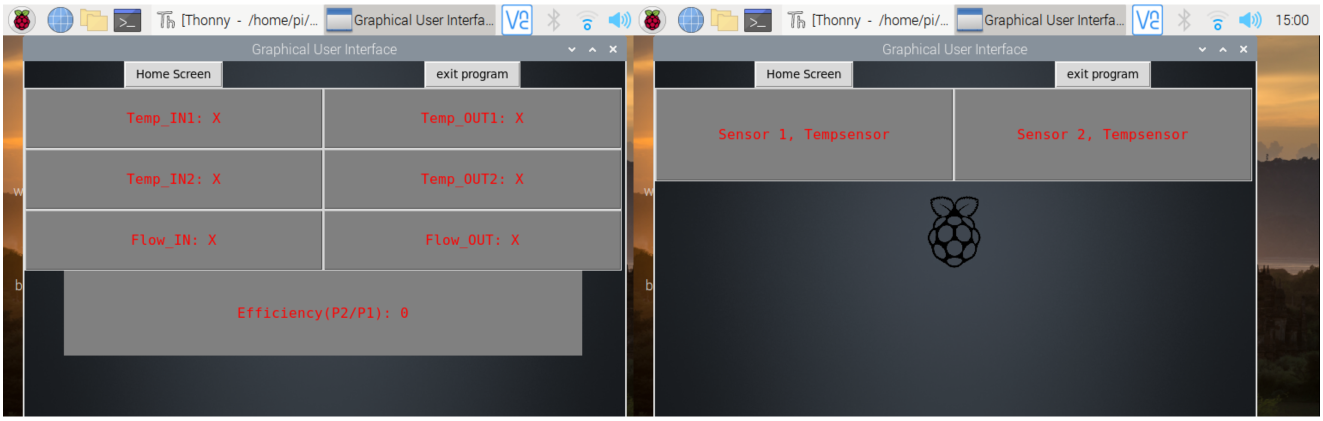


Figure 4.4: Layout of the finished GUI home screen



(a) GUI layout: Zone 1, showing data representation from the thermal system

(b) GUI layout: Zone 2, showing the additional zone 2 further development

4.3 Results and Testing: Data Transmission: MQTT

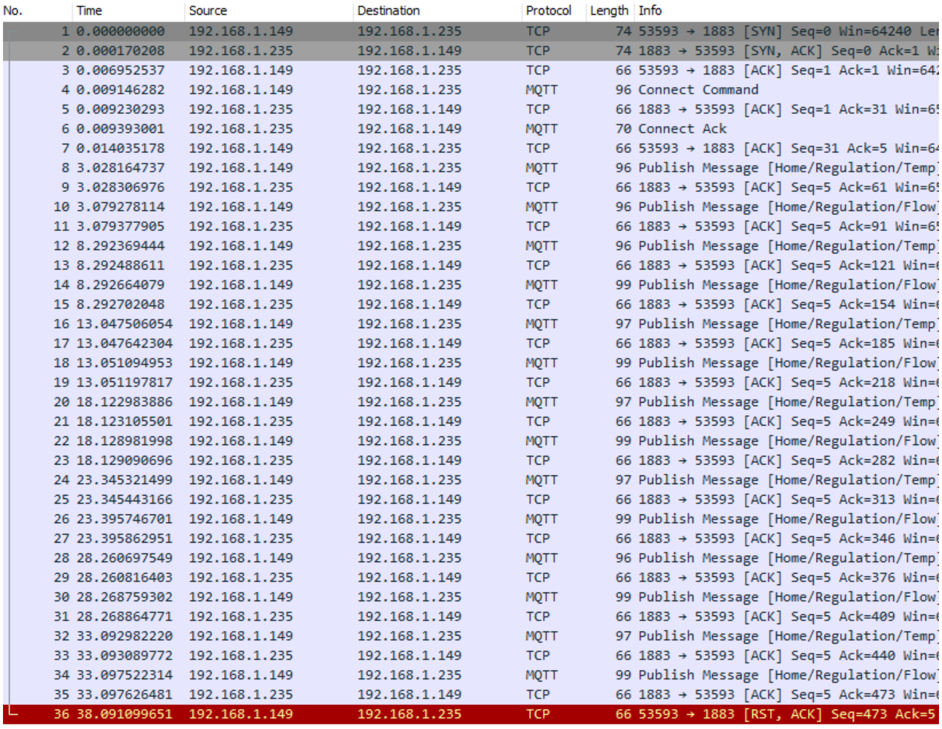
Testing the MQTT communication it is used Wireshark to analyze factors such as data throughput, latency, and data loss. Tolerance of relevant factors should not exceed the requirement of the system, this could cause harm to the system equipment.

4.3.1 Testing and Analysis

Wireshark has two filter mechanisms divided into two processes. The first filter is called *capture filter* and is used to filter out when capturing packets, e.g from a certain IP address.

After capturing a certain transmission flow based on certain defined factors the *display filter* can be used. The display filter is used to filter out based on new factors, e.g if the capture filter is filtering based on a certain IP address.
The display filter is not used due to the circumstances that the whole MQTT process is in interest.

By using capture filter and the IP address of the Broker(RPi hub: 192.168.1.235) the entire communication of the MQTT process can be captured. Packets shown in 4.6 are processed and shown in chronological order from top to bottom conformity to time.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.149	192.168.1.235	TCP	74	53593 → 1883 [SYN] Seq=0 Win=64240 Le
2	0.000170208	192.168.1.235	192.168.1.149	TCP	74	1883 → 53593 [SYN, ACK] Seq=0 Ack=1 W
3	0.006952537	192.168.1.149	192.168.1.235	TCP	66	53593 → 1883 [ACK] Seq=1 Ack=1 Win=64
4	0.009146282	192.168.1.149	192.168.1.235	MQTT	96	Connect Command
5	0.009230293	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=1 Ack=31 Win=6
6	0.009393001	192.168.1.235	192.168.1.149	MQTT	70	Connect Ack
7	0.014035178	192.168.1.149	192.168.1.235	TCP	66	53593 → 1883 [ACK] Seq=31 Ack=5 Win=6
8	3.028164737	192.168.1.149	192.168.1.235	MQTT	96	Publish Message [Home/Regulation/Temp]
9	3.028306976	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=61 Win=6
10	3.079278114	192.168.1.149	192.168.1.235	MQTT	96	Publish Message [Home/Regulation/Flow]
11	3.079377905	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=91 Win=6
12	8.292369444	192.168.1.149	192.168.1.235	MQTT	96	Publish Message [Home/Regulation/Temp]
13	8.292488611	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=121 Win=
14	8.292664079	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
15	8.292702048	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=154 Win=
16	13.047506054	192.168.1.149	192.168.1.235	MQTT	97	Publish Message [Home/Regulation/Temp]
17	13.047642304	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=185 Win=
18	13.051094953	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
19	13.051197817	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=218 Win=
20	18.122983886	192.168.1.149	192.168.1.235	MQTT	97	Publish Message [Home/Regulation/Temp]
21	18.123105501	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=249 Win=
22	18.128981998	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
23	18.129090696	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=282 Win=
24	23.345321499	192.168.1.149	192.168.1.235	MQTT	97	Publish Message [Home/Regulation/Temp]
25	23.345443166	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=313 Win=
26	23.395746701	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
27	23.395862951	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=346 Win=
28	28.260697549	192.168.1.149	192.168.1.235	MQTT	96	Publish Message [Home/Regulation/Temp]
29	28.260816403	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=376 Win=
30	28.268759302	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
31	28.268864771	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=409 Win=
32	33.092982220	192.168.1.149	192.168.1.235	MQTT	97	Publish Message [Home/Regulation/Temp]
33	33.093089772	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=440 Win=
34	33.097522314	192.168.1.149	192.168.1.235	MQTT	99	Publish Message [Home/Regulation/Flow]
35	33.097626481	192.168.1.235	192.168.1.149	TCP	66	1883 → 53593 [ACK] Seq=5 Ack=473 Win=
36	38.091099651	192.168.1.149	192.168.1.235	TCP	66	53593 → 1883 [RST, ACK] Seq=473 Ack=5

Figure 4.6: Wireshark: MQTT analysis

Analyzing the packet flow between the broker and the client, the three phases of the MQTT process can be distinguished. A more close up on what type of packets are transferred between the client and the broker is shown in Figure 4.7.

TCP Connection Establishment

The first three packets are the TCP Connection Establishment phase in the MQTT is the three-way handshake. The broker is the default in listening mode, where the first packet is sent from the client to the broker which is a communication request packet [SYN]. The second packet is where the broker responds with an ACK-packet to the received SYN-packet

[SYN, ACK]. The third packet is where the client confirms that it received the [SYN, ACK]-packet from the broker.

The TCP connection establishment in the project maintains the theoretical MQTT process presented in Subsection 2.3.3.

MQTT Connection Establishment

Packet 4 to 7 is where the Client initiates the MQTT connection with a connect package, as explained in Subsection 2.3.3 it contains necessary information about the client and its configuration towards MQTT‘s features. The broker responded with an ACK that the client connection packet has been received, and further a CONNACK where the configuration towards MQTT and client information has been validated as error-free.

packet 8 to 35 is the main process where the message exchange is being processed. The exchange shows only where 2 of 6 sensors are present, this was to reduce the number of packages being transferred to make it easier to analyze. Present sensors are one flow meter and one temperature meter with one topic each. This reduction has not affected the performance as the system maintains a load far under what the project devices and protocol can process.

The packages maintains an interval of approximate 5 seconds on each topic as presented code in appendix C and implementation Subsection 3.3.3.

The MQTT connection establishment and packet exchange maintains the theoretical functionality and practical execution of the project requirement presented in Subsection 2.3.3 and execution in Appendix C.

TCP Connection Termination

To progress further with a more interesting termination, its been chosen to take a more fatal ending process than engaging a normal symmetric three-way TCP termination sequence where TCP FIN is used. In this case, the client disconnects from the establishment because of a fatal process, e.g program error. Packet 30 dictates that the client is disconnected and the channel is now terminated, this is not a usual TCP event. The packet called RST (or Reset) is a flag within the TCP packet that ended the TCP connection immediately.

In this case, the TCP connection took an abrupt ending but maintains the theoretical functionality of a TCP connection. Further discussion towards this case is presented in Section 5.1.

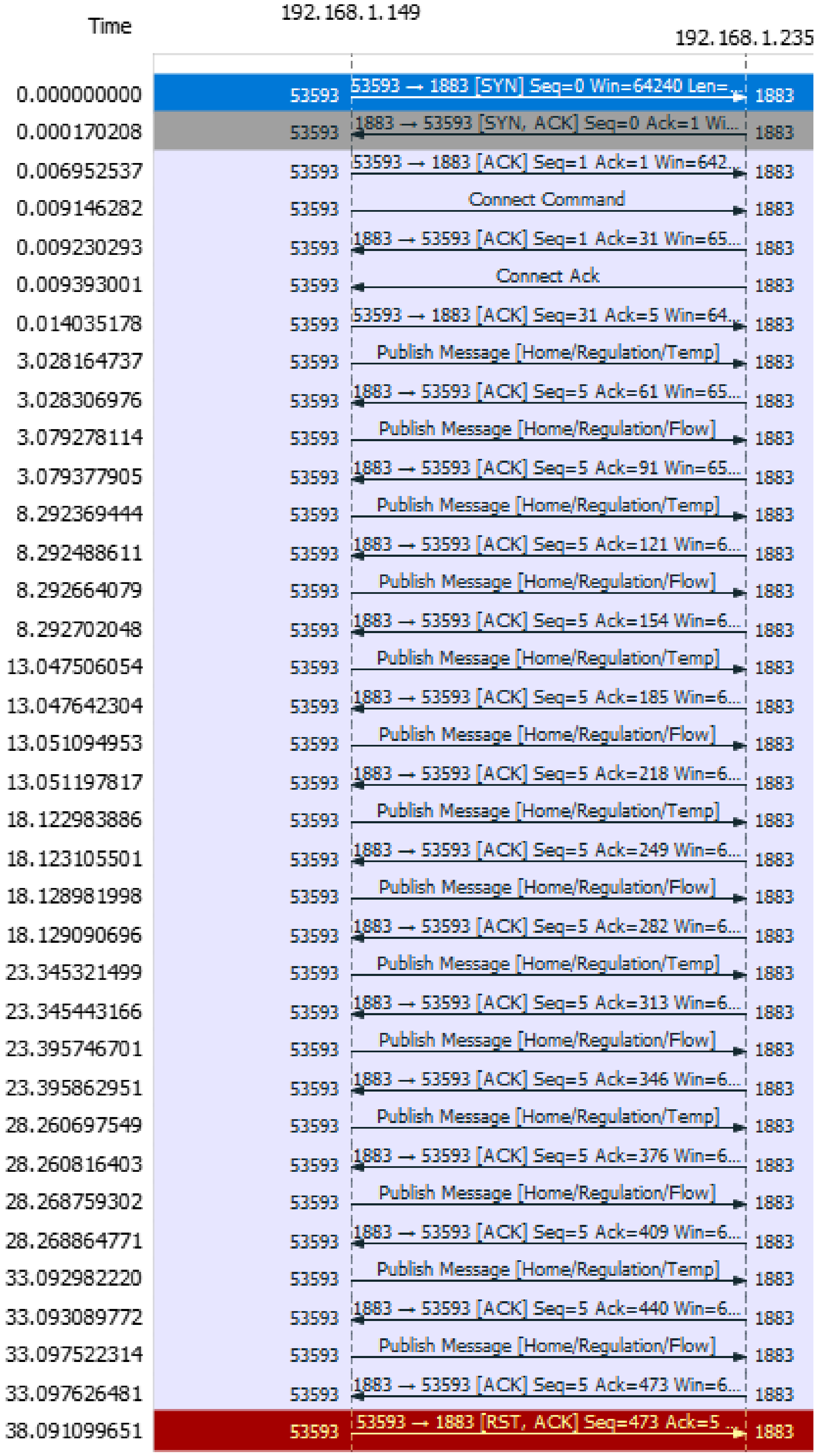


Figure 4.7: Wireshark: MQTT, packet exchange between client(192.168.1.149) and server/broker (192.168.1.235)

4.4 802.11ah

This section result is based on existing research by Silex Technology, cause of an unfortunate incident with the testing module.
Silex Technology used the same evaluation kit as presented in this thesis.

4.4.1 Implementation Error

Figure 4.8 presents the error code received during the implementation phase for one of the evaluation kits.

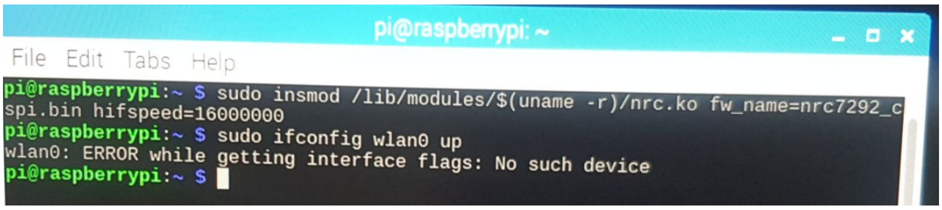


Figure 4.8: Error that occurred during implementation

4.4.2 802.11ah Range

The protocol's available power and functions used in its modules make the Wi-Fi HaLow competitive against other low-power wide-area networks (LPWAN) technologies like LoRa, SigFox, etc.
Based on the formula from 3.4.6 it is possible to make an assumption of the range of the protocol with certain equipment and transmit power. Considering the range calculation from the project, it requires more than Friis Transmission Formula and Fresnel Zone to estimate the certain range, as the real world consists of non-free paths. It is also important to consider the protocol restrictions for handling events such as packet loss and latency.

Silex used the same evaluation kit as the project planned to use to test the protocol with communication restrictions within Norway. Silex used the Iperf tool for testing, which included having modules set as an access point and a station mode and then set the available parameters to realistic values.
At 1 MHz wide channel width Silex achieved a throughput of 1.25 Mb/s, increasing the channel width to 4MHz wide it was an increase of 2,15 Mb/s throughput using bi-directional TCP traffic.

Silex also experienced an effect of decreased range when moving out of the Fresnel Zone, which means that it is important to consider the Fresnel Zone when designing a communication system.

1000 m	Bandwidth 1MHz (922.5MHz)	UDP	AP->STA	1.76	1.69	1.75	1.73
			STA->AP	1.39	1.35	1.31	1.35
		TCP	AP->STA	1.68	1.64	1.62	1.65
			STA->AP	1.35	1.31	1.42	1.36
	Bandwidth 2MHz (925MHz)	UDP	AP->STA	3.30	3.49	3.45	3.41
			STA->AP	3.00	3.16	3.25	3.14
		TCP	AP->STA	2.51	2.34	2.51	2.45
			STA->AP	2.18	2.13	2.10	2.14
	Bandwidth 4MHz (922MHz)	UDP	AP->STA	5.79	5.68	5.88	5.78
			STA->AP	4.92	5.60	5.50	5.34
		TCP	AP->STA	3.26	3.28	3.47	3.34
			STA->AP	3.12	2.99	2.93	3.01
1600 m	Bandwidth 1MHz (922.5MHz)	UDP	AP->STA	1.700	1.730	1.780	1.74
			STA->AP	1.570	1.610	1.470	1.55
		TCP	AP->STA	1.290	1.330	1.290	1.30
			STA->AP	1.220	1.210	1.210	1.21
	Bandwidth 2MHz (925MHz)	UDP	AP->STA	2.22	2.16	2.19	2.19
			STA->AP	2.53	2.39	2.49	2.47
		TCP	AP->STA	1.61	1.69	1.58	1.63
			STA->AP	1.74	1.71	1.67	1.71
	Bandwidth 4MHz (922MHz)	UDP	AP->STA				
			STA->AP				
		TCP	AP->STA				
			STA->AP				

Figure 4.9: Silex summary data from the testing [10]

Chapter 5

Conclusion and Future Work

5.1 Discussion and Recommendations

Based on the experience achieved in this project, some additional implementation measures can be performed regarding security, availability, and functional up-time.

Data collection for thermal water supply system

Although the system of data collection and data handling system core function is designed, implemented, and tested, the system is fully capable of enhancements. During the design phase, it was chosen an optional sector of the system to implement a connection to the Internet, causing the system to fall under the IoT category to bring additional functionality to the system. Further work will include implementing this connection into the system. Also, an additional zone is presented in the GUI, that can be used to further control the thermal supply system when considering control functions.

Also adding a data logger that logs the sensor data is highly prioritized for further work. Implementing and designing a data logger and a database to the system helps the user of the system to access historical data to analyze the operative process over time. Showing only present data on the GUI will not get the perspective of how the system operates during a certain period. Showing data gathered over a week could help the user to distinguish system error(s) or inefficiency of the system over a certain time of the day. Further, the user could make enhancements to the system to make it more efficient or prevent damage to the equipment.

Data Transmission: MQTT

During the design and implementation is was presented some information about security and

additional features that is available in MQTT, but not implemented into the system. Considering security measurements, it is recommended to use authentication and authorization to secure the communication between the broker and the clients, especially infrastructure that transfers sensitive and critical information that can be used by others to cause harm.
Further work will be to utilize the security function of the MQTT protocol to improve the overall security of the system. Also, utilize MQTT features such as last will and testament if further development is implemented to the system.
As the case in Subsection 4.3.1 regarding the TCP connection termination where the connection took an abrupt ending. Further work can be to configure and implement *Clean Session* and *Last Will* that includes in MQTT features explained in Subsection 2.3.4. This will make preparations towards fail-safe functionality if the system should be further developed to handle more crucial applications or control mechanisms.

Wi-Fi HaLow 802.11ah

The theory around protocol development and technologies has a broad architecture that is meant to solve technical issues in today's IoT challenges but seems to be a complex "fight" in our society. That is an understandable challenge for society to present the "best" protocol when the broad availability of the protocols is highly dependent on the specific usage. As a self develop thought considering the objective **Overview and Test 802.11ah** it would be interesting to see its potential and change when its implementation has begun in the devices and networks of the market.
When it comes to Wi-Fi HaLow the theoretical specification and existing research Wi-Fi HaLow specifications meet the requirements for IoT applications. The protocol brings also forward new interesting features and mechanisms that will solve many nowadays advancing requirements and open up for new system architectures and applications.
Comparing Wi-Fi HaLow to other communication protocols in the LPWAN domain in similar applications in many cases it exceeds the other protocols in the specific IoT requirements such as energy efficiency, range, and data rate. Based on the formula from 3.4.6 it does fulfill the explanation for how long the range for Wi-Fi HaLow, but does not explain how the protocol handles packet loss and re-transmissions if a packet is lost due to some circumstances. When designing a communication system its important to take into account Friis formula for antenna theory and the Fresnel zone, it can be used to acquire much higher overall performance considering range and data rate.

5.2 Conclusions and Contributions

Due to the circumstances (the COVID-19 pandemic), which is still a lockdown of many sectors of our society from 2020. Working alone in this project has been challenging due to lack of cooperation with other students and absence from the school environment.

The primary objective of this thesis was to develop a data collection system for a thermal water supply system with an MQTT-based transmission system, and further test the Wi-Fi HaLow protocol to present its functionality in a realistic environment that considers its technical specification and its theoretical aspect. Considering the result acquired during the test phase it is to consider that the primary core of the data systems objective have been successfully, but has potential for further development which would be presented in the next Section.

When it comes to Wi-Fi HaLow, it does fulfill the requirement for IoT and is gone through several tests and simulations based on the theoretical specification of the protocol. It is still an early phase for the protocol and its implementation procedure to the existing network is interesting, but it takes to hold on to testing a finished product in an IoT environment for further enhancement.

Wi-Fi HaLow‘s theoretical representation is easily accessed through websites and companies like Wi-Fi alliance and IEEE, but testing the protocol in a physical environment in an early state is rather challenging for the private sector. Answering the question "Why didn't the 802.11ah module work? and why couldn't testing be performed?" would be that the testing environment isn't arranged or have the focus for the student/private sector to be performed. The project Wi-Fi module bears the mark of a new test module, as the module is dependent on an existing module to operate.

Considering the findings of the protocol based on the existing research from the report from Silex Technology, it concludes that the protocol can achieve a stable connection for over 3km if the design is done correctly by considering the application operative area. The range of transport protocols doesn't only depend on the protocol itself, but on the available power the transport protocol has to spare and the technical specification of the antenna used.

Throughout the project, it has been experienced that a more narrow focus would be more efficient for achieving higher acquisitions instead of a broad specter of protocol environments and solutions.

In overall conclusion for the three objectives of this thesis, the project is considered successful and educational for the thesis member and hopefully for its intended readers.

5.3 Future Work

Future work of the data collection and transmission system is to consider the measurements discussed in 5.1 to further enhance the system functionality and user experience. This requires further testing towards the new functions, but will robust system that can handle unexpected events that can occur.
Further work regarding testing of the 802.11ah protocol includes continuing troubleshooting of the received hardware with Silex Technology to be able to perform an analysis towards the protocol functionality and performance.

References

[1] 4G Americas, "Cellular Technologies Enabling the Internet of Things," 2015. https://www.5gamericas.org/wp-content/uploads/2019/07/4G_Americas_Cellular_Technologies_Enabling_the_IoT_White_Paper_-_November_2015.pdf [Accessed on 05.03.2021].

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[3] M.Pathak, "Overview of an IoT Protocols ," 2019. https://www.linkedin.com/pulse/overview-iot-protocols-maneesh-pathak/ [Accessed on 03.06.2021].

[4] MQTT.org, "Why MQTT?," 2020. https://mqtt.org/ [Accessed on 03.06.2021].

[5] BytesofGigabytes, "MQTT Protocol Handshake," 2020. https://bytesofgigabytes.com/mqtt/mqtt-protocol-handshake/ [Accessed on 26.05.2021].

[6] The HiveMQ Team, "MQTT Client and Broker and MQTT Server and Connection Establishment Explained," 2019. https://www.hivemq.com/blog/mqtt-essentials-part-3-client-broker-connection-establishment/ [Accessed on 03.06.2021].

[7] K. Tempfli, W. Bakker, G. C Huurneman, L.L.F Janssen, "Principles of remote sensing : an introductory textbook," 2009. https://www.researchgate.net/publication/233793398_Principles_of_remote_sensing_an_introductory_textbook [Accessed on 03.06.2021].

[8] K. Qian and M.Wang, "A look at IEEE 802.11AH," 2015. https://www.mwee.com/content/look-ieee-80211ah-wi-fi-below-1-ghz/page/0/1 [Accessed on 09.03.2021].

[9] Silex Technology, “How does 802.11ah compare with other LPWAN technologies?,” 2020. https://www.silextechnology.com/hubfs/White_Papers/SX-NEWAH%20White%20Paper_FINAL2020_v8.pdf [Accessed on 10.03.2021].

[10] Silex Technology, “How far can 802.11ah can go before the connection is ’Dead’?,” 2020. https://www.silextechnology.com/unwired/how-far-can-802.11ah-can-go-before-the-connection-is-dead [Accessed on 14.05.2021].

[11] S. E. R. AS, *Varmepumper bygningsoppvarming*. SINTEF Energy Research AS, 1997.

[12] L.An Phan and T.Kim, “Breaking Down the Compatibility Problem in Smart Homes: A Dynamically Updatable Gateway Platform,” 2020. https://www.mdpi.com/1424-8220/20/10/2783 [Accessed on 30.05.2021].

[13] W.Chai, “Smart home hub (home automation hub),” 2021. https://internetofthingsagenda.techtarget.com/definition/smart-home-hub-home-automation-hub [Accessed on 30.05.2021].

[14] J.Callaham, “What is a smart home, and why should you want one?,” 2021. https://www.androidauthority.com/what-is-a-smart-home-806483/ [Accessed on 23.01.2021].

[15] N.Tesla, “When woman is boss,” 1926. http://www.tfcbooks.com/tesla/1926-01-30.htm [Accessed on 22.02.2021].

[16] B.Russel and D. Duren, *Practical Internet of Things Security*. PACKT publishing, 2016.

[17] K. Shafique, B. A. Khawaja, F. Sabir, S. Qazi, and M. Mustaqim, “Internet of things (iot) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5g-iot scenarios,” *IEEE Access*, vol. 8, pp. 23022–23040, 2020.

[18] M.Burmester, “Home, smart home theory and practice of the connected home,” 2017. https://www.uid.com/en/news/home-smart-home-theory-and-practice-of-the-connecte [Accessed on 05.03.2021].

[19] OASIS Open, “MQTT Version 5.0,” 2019. https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf [Accessed on 03.06.2021].

[20] HiveMQ GmbH, “Publish Subscribe,” 2015. https://www.hivemq.com/blog/mqtt-essentials-part2-publish-subscribe/ [Accessed on 26.11.2020].

[21] HiveMQ GmbH, "Wildcards," 2019. https://www.hivemq.com/blog/mqtt-essentials-part-5-mqtt-topics-best-practices/ [Accessed on 26.11.2020].

[22] SolarWinds Worldwide LLC, "Understanding QoS and Its Importance in Network Monitoring," 2020. https://www.tek-tools.com/network/qos-monitoring-tools [Accessed on 30.05.2021].

[23] The HiveMQ Team, "Quality of Service," 2015. https://www.hivemq.com/blog/mqtt-essentials-part-6-mqtt-quality-of-service-levels/ [Accessed on 30.05.2021].

[24] The HiveMQ Team, "Retained Messages," 2015. https://www.hivemq.com/blog/mqtt-essentials-part-8-retained-messages/ [Accessed on 30.05.2021].

[25] M. de Vivo, G.O. de Vivo, R.Koeneke, G.Isern, "Internet Vulnerabilities Related to TCP/IP and T/TCP," 1999. https://dl.acm.org/doi/pdf/10.1145/505754.505760 [Accessed on 30.05.2021].

[26] The HiveMQ Team, "The problem of half-open TCP connections," 2015. https://www.hivemq.com/blog/mqtt-essentials-part-10-alive-client-take-over/ [Accessed on 30.05.2021].

[27] The HiveMQ Team, "Persistent Session," 2015. https://www.hivemq.com/blog/mqtt-essentials-part-7-persistent-session-queuing-messages/ [Accessed on 30.05.2021].

[28] The HiveMQ Team, "Last Will and Testament," 2015. https://www.hivemq.com/blog/mqtt-essentials-part-9-last-will-and-testament/ [Accessed on 31.05.2021].

[29] The HiveMQ Team, "HiveMQ - MQTT Security Fundamentals," 2015. https://www.hivemq.com/blog/introducing-the-mqtt-security-fundamentals/ [Accessed on 09.05.2021].

[30] Wi-Fi Alliance, "Wi-Fi HaLow," 2020. https://www.wi-fi.org/discover-wi-fi/wi-fi-halow [Accessed on 28.1.2021].

[31] L. Tian, E. Lopez-Aguilera, E. Garcia-Villegas, M. T. Mehari, E. De Poorter, S. Latré, and J. Famaey, "Optimization-oriented raw modeling of ieee 802.11ah heterogeneous networks," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10597–10609, 2019.

[32] S. Santi, L. Tian, E. Khorov, and J. Famaey, "Accurate energy modeling and characterization of ieee 802.11ah raw and twt," *Sensors*, vol. 19, 06 2019.

[33] B.Ray, "Licensed Vs. Unlicensed Spectrum: What's The Difference?," 2020. https://www.iotacommunications.com/blog/licensed-vs-unlicensed-spectrum/ [Accessed on 09.03.2021].

[34] E.Marske, "Which Unlicensed Frequency is Best for Your Industrial Wireless Network?," 2019. https://www.esteem.com/2019/01/unlicensed-frequencies-for-industrial-wireless-networks/ [Accessed on 09.03.2021].

[35] P.Vouzis, "What is the MCS Index?," 2020. https://netbeez.net/blog/what-is-mcs-index/ [Accessed on 13.04.2021].

[36] Thermometrics Corporation, "PT-1000 Sensor," 2011. https://www.thermometricscorp.com/pt1000 [Accessed on 11.05.2021].

[37] Sequent Microsystems, "Mega-RTD eight channel," 2020. https://cdn.shopify.com/s/files/1/0176/3274/files/Mega-RTD-UsersGuide.pdf?v=1604074029 [Accessed on 11.05.2021].

[38] IEC, "IEC.60751," 2008. https://www.standard.no/nettbutikk/produktkatalogen/produktpresentasjon/?ProductID=339942 [Accessed on 02.06.2021].

[39] IBM Cloud Education, "Message Brokers," 2020. https://www.ibm.com/cloud/learn/message-brokers [Accessed on 03.06.2021].

[40] Oracle Corporation, "Message-Oriented Middleware (MOM)," 2010. https://docs.oracle.com/cd/E19340-01/820-6424/aeraq/index.html [Accessed on 03.06.2021].

[41] G. Combs, "Display Filters," 2017. https://www.wireshark.org/docs/man-pages/wireshark-filter.html [Accessed on 02.06.2021].

[42] Silex Technology, "SX-NEWAH Evaluation," 2021. https://www.silextechnology.com/connectivity-solutions/embedded-wireless/sx-newah-evaluation [Accessed on 07.04.2021].

[43] NKOM, "Om Nkom," 2021. https://www.nkom.no/om-nkom [Accessed on 23.01.2021].

[44] ETSI, “ETSI TR 103 245,” 2014. https://www.etsi.org/deliver/etsi_tr/103200_103299/103245/01.01.01_60/tr_103245v010101p.pdf [Accessed on 01.03.2021].

[45] European Commission, “CE marking,” 2021. https://ec.europa.eu/growth/single-market/ce-marking_en [Accessed on 03.03.2021].

[46] The Raspberry Pi Foundation, “Product compliance and safety,” 2021. https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md [Accessed on 03.03.2021].

[47] Silex Technology America, “APPLICATION NOTE: SX-NEWAH Channel Mapping,” 2020. https://www.silextechnology.com/hubfs/Application%20Notes/142-20159-100A_AppNote_SX-NEWAHChannelMapping_June2020_142-20142-100.pdf [Accessed on 03.03.2021].

[48] IEEE, “145-1983 - IEEE Standard Definitions of Terms for Antennas,” 1983. https://ieeexplore.ieee.org/document/30651 [Accessed on 03.03.2021].

[49] Editorial Team - everything RF, “What is the Fresnel Zone?,” 2018. https://www.everythingrf.com/community/what-is-the-fresnel-zone [Accessed on 14.05.2021].

Appendix A

GUI, RPi3

```
from tkinter import * #
from tkinter import font
from PIL import ImageTk, Image
import tkinter as tk

#Files:
from MQTT_Subscribe import *

#Window config
Home_screen = Tk()
Home_screen.title("Graphical User Interface") #Program Title
#Home_screen.iconbitmap("weather_icon.ico") #Program Icon
Home_screen.geometry("700x550") #Screen size of RPi

#Window background
def on_resize(event):
  # resize the background image to the size of label
  image = bgimg.resize((event.width, event.height), Image.ANTIALIAS)
  # update the image of the label
  bg_label.image = ImageTk.PhotoImage(image)
  bg_label.config(image=bg_label.image)

bgimg = Image.open('background.png') # load the background image
bg_label = Label(Home_screen)
```

```
bg_label.place(x=0, y=0, relwidth=1, relheight=1) # make label l to fit the parent window always
bg_label.bind('<Configure>', on_resize) # on_resize will be executed whenever label l is resized

global my_image_label

def Zone_1_technical():
    #remove others:
    Zone_1.grid_forget()
    Zone_2.grid_forget()

    #Restore/create:
    Button_home_screen.grid(row=0, column=0)
    Button_temp1_in.grid(row=1, column=0)
    Button_temp2_in.grid(row=2, column=0)
    Button_temp1_out.grid(row=1, column=1)
    Button_temp2_out.grid(row=2, column=1)
    Button_flow_in.grid(row=3, column=0)
    Button_flow_out.grid(row=3, column=1)
    Label_efficiency.grid(row=4, column=0, columnspan=2)

def Zone_2_livingroom():
    #remove others:
    Button_home_screen.grid(row=0, column=0)
    Zone_1.grid_forget()
    Zone_2.grid_forget()

    #Restore/create:

    Button_temp.grid(row=1, column=0)
    Button_humidity.grid(row=1, column=1)

def Back_home_screen():
    #remove others:
    Button_temp.grid_forget()
    Button_humidity.grid_forget()
    Button_temp1_in.grid_forget()
    Button_temp2_in.grid_forget()
```

```
    Button_temp1_out.grid_forget()
    Button_temp2_out.grid_forget()
    Button_flow_in.grid_forget()
    Button_flow_out.grid_forget()
    Label_efficiency.grid_forget()
    Button_home_screen.grid_forget()

    ##Restore/create:
    Zone_1.grid(row=1, column=0)
    Zone_2.grid(row=1, column=1)

#Labels:
Label_efficiency = Label(Home_screen, text="Efficiency(P2/P1): ",font=("Terminal"))

#Buttons:
Zone_1    = Button(Home_screen, text="Zone 1 Regulation" , font=("Terminal"), height=10, width=32,
fg="red",bg="grey", command = Zone_1_technical)
Zone_2    = Button(Home_screen, text="Zone 2 living room", font=("Terminal"), height=10, width=32, fg="red",bg="grey", command = Zone_2_livingroom)

#Zone 1:
Button_temp1_in = Button(Home_screen, text="Temp_IN1: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Button_temp2_in = Button(Home_screen, text="Temp_IN2: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Button_temp1_out = Button(Home_screen, text="Temp_OUT1: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Button_temp2_out = Button(Home_screen, text="Temp_OUT2: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Button_flow_in = Button(Home_screen, text="Flow_IN: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Button_flow_out = Button(Home_screen, text="Flow_OUT: X", font=("Terminal"), height=3, width=32, fg="red",bg="grey")
Label_efficiency = Label(Home_screen, text="Efficiency(P2/P1): %d" %message_received ,font=("Terminal"), height=5, width=60, fg="red",bg="grey")

```

```
#Zone 2: (For further development)
Button_temp = Button(Home_screen, text="Sensor 1, Tempsensor", font=("Terminal"), height=5, width=32,
fg="red",bg="grey")
Button_humidity = Button(Home_screen, text="Sensor 2, Tempsensor", font=("Terminal"), height=5, width=32,
fg="red",bg="grey")

#General program buttons
Button_quit = Button(Home_screen, text="exit program", command=Home_screen.destroy)
Button_home_screen = Button(Home_screen, text="Home Screen", command=Back_home_screen)

#Grid system:
Button_quit.grid(row=0, column=1)
Zone_1.grid(row=1, column=0)
Zone_2.grid(row=1, column=1)

#Main loop:
Home_screen.mainloop()
```

Appendix B

MQTT Subscribe, RPi3

```
import paho.mqtt.client as mqtt
from time import sleep

global message_received
message_received = 0
def on_connect(client, userdata, flags, rc):
  print(f"Connected with result code {rc}")

  client.subscribe("Home/Regulation/#") #Subscribes to all topics
                                        #within "Regulation"

  #client.subscribe("Home/Regulation/Source_temp_out")
  #client.subscribe("Home/Regulation/Source_flow_out")

  #client.subscribe("Home/Regulation/Heat_temp_In")
  #client.subscribe("Home/Regulation/Heat_temp_out")
  #client.subscribe("Home/Regulation/Heat_flow_out")

# the callback function, Triggered when a messages is received
def on_message(client, userdata, msg):
  print(f"{msg.topic} {msg.payload}")
  global message_received
  sleep(1)
  message_received = msg.payload

client = mqtt.Client()
```

```
client.on_connect = on_connect
client.on_message = on_message
client.will_set('raspberry/status', b'{"status": "Off"}')

# Client connection; Three parameters broker address, broker port number, and
  keep-alive time respectively
client.connect("127.0.0.1", 1883, 60)

# set the network loop block
client.loop_start()
```

Appendix C

Code: Regulation w/MQTT client, RPi Zero

```
import RPi.GPIO as GPIO
import megabas
import time
import threading
import paho.mqtt.client as mqttClient
from time import sleep


broker_address = "192.168.1.235"
client = mqttClient.Client("Rasp_zero_client")
client.connect(broker_address, port =1883)
sleep(3)


Flow_sensor_1 = 11 # Flow sensor connected to pin 11
#Flow_sensor_2 = 13


GPIO.setmode(GPIO.BOARD)
GPIO.setup(Flow_sensor_1, GPIO.IN, pull_up_down = GPIO.PUD_UP) # Configure pin 11 for signals


global count_1 # Pulses registered
```

```
global flow_1 # Flow variable
count_1 = 0
flow_1 = 0

global R_1   # Resistance variable for PT-1000
global temp_1 # Temperatur variable (variates with R_1)
R_1 = 0

def Count_1(self):
  global count_1
  global flow_1
  count_1 = count_1+1        # +1 every time a pulse has been detected
  flow_1 = count_1 * 0.5/ (60) # Calculates flow from count_1
  flow_1 = round(flow_1,2)   # Rounds off to two numbers after comma

def reset():
  global flow_1
  global R_1
  global temp_1

  threading.Timer(5.0,reset).start() # Timer interval (set to 5 seconds)

  print("Measured flow:", flow_1)   # Prints flow value (just for visualization)

  R_1 = megabas.getRIn1K(0,1)       # Retrives resistance value from the PT-1000 element on card 0 channel 1
  print("Measured resistance:", R_1) # Prints the measured resistance (just for visualization)
  temp_1 = ((R_1*1000)-1000)/(3.851) # Converting resistance to degrees
  temp_1 = round(temp_1,2)
  print("Measured Temp:", temp_1)   # Prints the calculated temp from R (just for visualization)

  msg_temp_1 = "Temp:" + str(temp_1) # Creating mqtt message description and value for temp
  msg_flow_1 = "Flow:" + str(flow_1) # Creating mqtt message description and value for flow
```

```
    client.publish("Home/Regulation/Temp", msg_temp_1) #Publishing temp_1 value to topic Temp
    client.publish("Home/Regulation/Flow", msg_flow_1) #Publishing flow_1 value to topic Flow

    sleep(3)
    flow_1 = 0
    count_1 = 0
    print("RESET")


GPIO.add_event_detect(Flow_sensor_1, GPIO.FALLING, callback=Count_1) #Detects pulses on pin 11

reset()
```