DEVELOPMENT OF AFFORDABLE RAIN GAUGE FOR RAIN RATE AND FADE MEASUREMENTS

CHUAH CHONG HAO

UNIVERSITI TUNKU ABDUL RAHMAN

DEVELOPMENT OF AFFORDABLE RAIN GAUGE FOR RAIN RATE AND FADE MEASUREMENTS

CHUAH CHONG HAO

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering Electrical and Electronic Engineering with Honours

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

April 2024

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature :

Name : Chuah Chong Hao

ID No. : 1901470

Date : 26 April 2024

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"DEVELOPMENT OF AFFORDABLE RAIN GAUGE FOR RAIN RATE AND FADE MEASUREMENTS"** was prepared by **CHUAH CHONG HAO** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Engineering Electrical and Electronic Engineering with Honours at Universiti Tunku Abdul Rahman.

Approved by,

Signature : 
____

Supervisor : Dr Hafiz Basarudin

Date : 28/04/2024

Signature : _________________________________

Co-Supervisor : _________________________________

Date : _________________________________

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. First of all, I would like to express my appreciation to my research supervisor, Dr. Hafiz bin Basarudin for his invaluable time of providing his best guidance and being patient throughout the development of this project. His guidance and dedication into this project progress has provided to the smooth success of the execution of the research project and I have gained a lot of new knowledge from him.

Next, I would also like to express my gratitude to my loving parents and friends who had helped me with necessary resources and provided me encouragement and motivation throughout the period of conducting this research project.

ABSTRACT

Rain rate and rain fade are important parameters that are used in the world of telecommunications. Rain rate and fade directly affect the signal strength and signal transmission of the communication systems. The rain fade is a useful parameter to observe signal loss on high frequency links due to rain. Rain rate is the key parameter to calculate rain fade. The measurement of rain rate is in the unit of millimeter per hour while the rain fade is in the unit of decibels per kilometer. The rain rate is usually determined by using a device called the rain gauge but current industrial grade instruments for measuring the rain rate are too expensive and not affordable. The data provided online is also inconsistent without frequent updates and require longer time to obtain them. The rain gauge available in the market also lacks the use of IoT, making obtaining the data to be in manual mode and require frequent human interaction with it. Hence, a low-cost microcontroller-based rain gauge that uses the tipping bucket mechanism is designed and developed in this project research. The rain gauge uses 3D printing materials, lowering the cost and also uses low-cost components to limit the total cost of the project. The electronic circuit of this project comprises of a Hall Effect Sensor, a ESP32 microcontroller, a real-time clock module and a MicroSD card module. The data is programmed to be stored in both Google Sheets and in MicroSD card as a backup at every 1 minute interval. The whole system is stored within a waterproof electrical junction box, sealed with silicone strip, providing an extra layer of waterproof to the circuit. The collected data is compared and analyse with reference to a nearby station that provides rainfall data and from the analysis, the difference is in an acceptable range.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

R	rainfall rate, mm/hr
A	rain fade, dB
a	frequency coefficient, GHz
b	frequency coefficient, GHz
γ_R	rain attenuation, dB/km
k	frequency constants, GHz
α	frequency constants, GHz
Z	reflectivity factor
A_T	horizontal path attenuation, dB
ITU	International Telecommunication Union
MANOBS	Manual of Surface Weather Observations
Z-R	Marshall-Palmer Relation
PLA	Polylactic Acid
PLA+	Polylactic Acid Plus
PETG	Polyethylene Terephthalate Glycol
ABS	Acrylonitrile Butadiene Styrene

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background and Importance of Study

Rainfall is a natural phenomenon that is very important across many different sectors of human activities. Human activities such as agriculture, hydrology, and environmental management requires the understanding and accurate measurement of rainfall, especially the intensity of the rainfall, or the rain rate, and its distribution. These data are significant in the decision-making for the stated fields. Another field that will be impacted by the rainfall is the wireless communication systems. The intensity of rainfall will directly cause the degradation of signals travelling through the rain, fading the signals. With the study of rainfall intensity or rain rate, the fading of signals or the rain fade can be obtained and the strategies to tackle this problem faced can be made and predicted when rain falls.

There are a lot of different types of rain gauges, with the most basic rain gauge requires manual emptying the tube before taking the data again. This rain gauge may be economically efficient, however may provide inaccurate data as when the tank is full, further measurements could not be taken.

Figure 1.1: Basic rain gauge (NASA, n.d.).

This type of rain gauge can also be made very easily with just a plastic water bottle with careful calibration for measurement of data. The below figure shows the homemade version of the rain gauge that uses the same measuring principle in Figure 1.1.

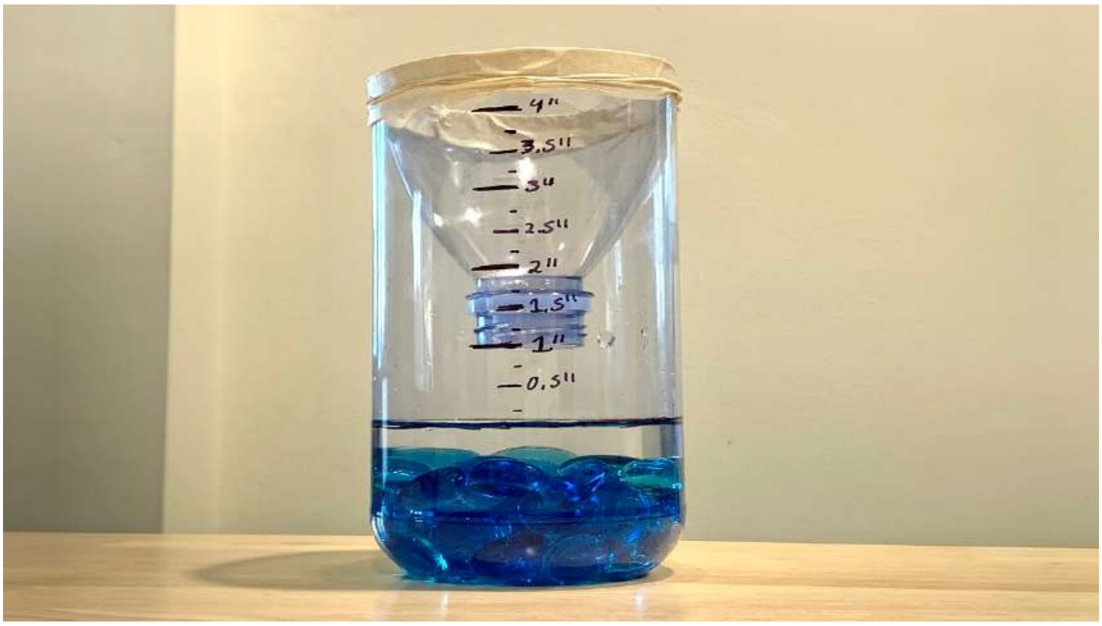

Figure 1.2: Homemade version of Rain Gauge (Freeman, 2021).

This type of rain gauge may be easy to be made, and can be deployed without a big budget, however, there is a limitation to this design as the tank can only hold that much volume of rain water. Hence, another type of rain gauge is proposed to be used, the tipping bucket rain gauge.

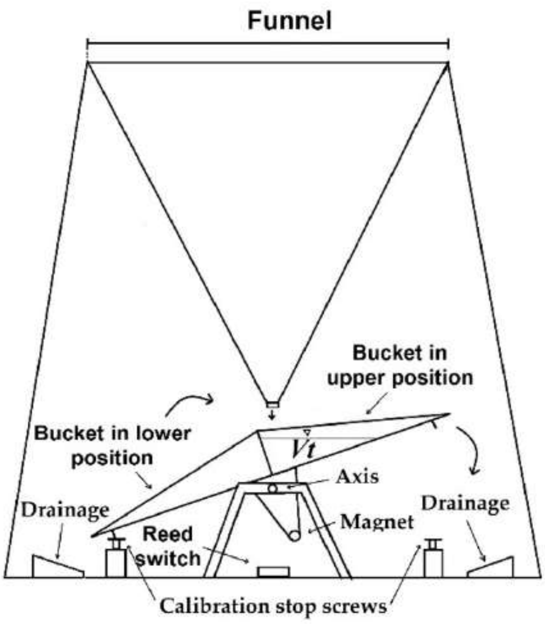

Figure 1.3: Tipping Bucket Rain Gauge (Segovia-Cardozo et al., 2023).

The tipping bucket rain gauge is widely used in rainfall monitoring as it is simple, consume low energy and very effective in measuring the rainfall (Segovia-Cardozo et al., 2023). The further discussion of various types of rain gauges will be continued in the next chapter.

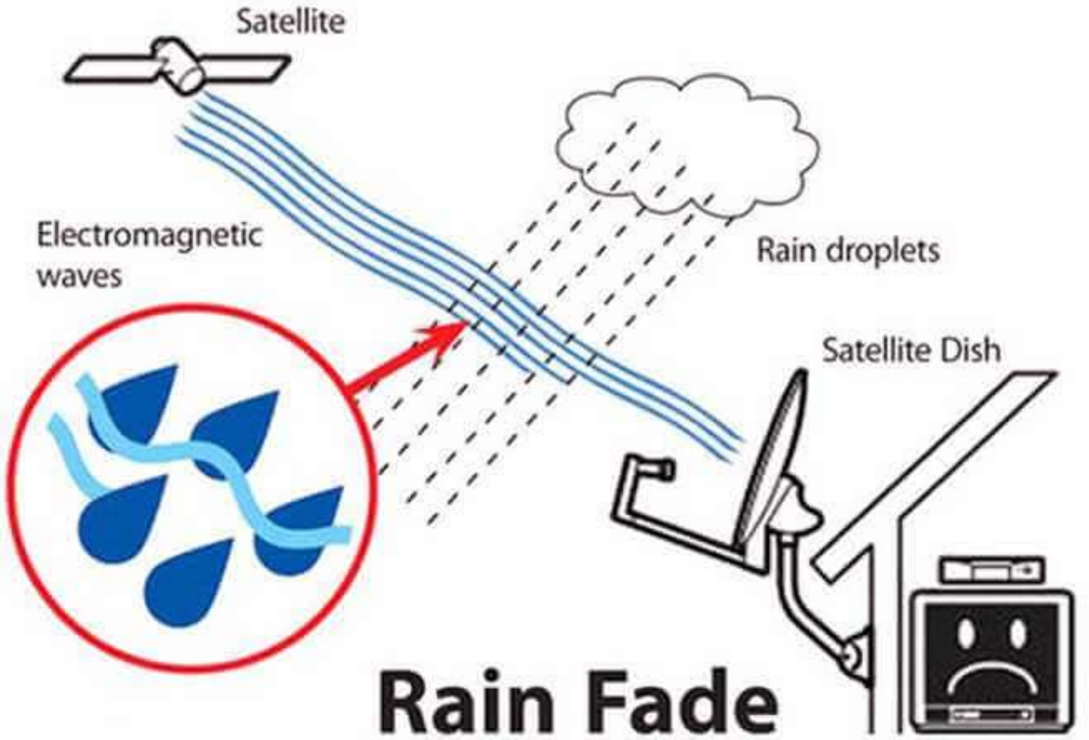

Figure 1.4: Rain fade illustration (Gannon, n.d.).

With the measurement of rain rate, the rain fade will be able to be obtained through equations. The rain fade is a parameter that is very important in the field of wireless communication. To explain briefly on rain fade, when the rain interact with the electromagnetic waves, the signal through the rain will be faded and degraded. The signal attenuation will be more obvious as the rain rate or rain intensity increases. This will cause the quality of wireless communication to be deteriorated. Hence, the real-time accurate measurements of the rain rate must be done to ensure that this problem can be solved. With the rain rate data, adaptive modulation and coding schemes can be implemented to adjust the transmission parameters to reduce the effect of rain fade.

1.2 Problem Statement and Identification

The main problem of the rain gauges is that it is generally very expensive. Although inexpensive models of the rain gauges are available at the market, the cheap rain gauge models provides lower accuracy of data and mostly require manual emptying the container that is used to store the rainwater. Inaccurate data of rainfall will also leads to the inaccurate data analysis of rain fade and hence causing the process of reduction of effect from rain fade in wireless communication to be ineffective which then leads to bigger problem such as increase in costing and loss of wireless communication. Next, the current available data from the International Telecommunication Union is also not up to date as the last updated date was on 20th June of 2017. This means that the current data may not be that accurate when used as the rainfall rate by each year may be affected by climate changes. Therefore, it is very crucial in reproducing or developing an affordable rain gauge to obtain the rain rate data to estimate the rain fade for the decision making at different applications that are impacted by the intensity of rain. The developing of a rain gauge is also important to provide real-time and up-to-date data for the estimation of rainfall rate.

1.3 Aims and Objectives

The main goal of this project is to produce and develop a low-cost or affordable microcontroller-based measuring device that has the function of measuring the rain rate and data-log the rain data obtained, and by using the data obtained, calculate the rain fade. To explain further, an affordable tipping bucket rain gauge will be developed and deployed for a time interval and the microcontroller will be responsible to obtain and logging the rain rate data constantly. These rain rate data collected will then be analysed and converted into rain fade data for the use of different fields. In summary, the main objectives and aims of the project is as below:

- To design and construct an affordable and low-cost rain rate measuring device that is able to withstand different environment for a long time in the aim of collecting the rain rate data for as long as possible.
- To develop a microcontroller-based tipping bucket rain gauge that is low-cost but is capable in obtaining the accurate and reliable rain rate data to obtain the trend of the rainfall rate for future study.
- To verify the accuracy of the data obtained by comparing the actual rain rate data from an industrial-grade tipping bucket rain gauge and data from online sources.

1.4 Scope and Limitation of the Study

In this project, a tipping bucket will be used as the device to measure the total rain rate readings. In the industry standards, stainless steel is used as the material for the rain gauge but in this project, a 3D-printed tipping bucket, funnel and mould will be used due to its lower cost than the stainless steel material. This is because the objective of this project is to be able to develop and deploy an affordable rain gauge. With the use of 3D printing material, differing from the standard material, the results obtained from the rain gauge may cause some difference and causes some inaccuracy. However, this claim can only be verified once the prototype is field-tested.

Furthermore, the construction of the prototype device will be made out of only a 3D printer without any special devices or machinery required. Due to that, the size of the prototype may be small in size and the precision of the device may need to be tested and calibrated as differ to the standard industrial-grade rain gauges. Next, the duration of the research study and the construction of the prototype is also highly limited to around 9 months (two trimesters), causing the duration of collecting the data to be shortened as the construction of the hardware prototype took a lot of time. Hence, in this project, the main scope is to be able to collect the data and visualize the trend of the data without too much of analysing the data as to analyse a dataset, at least one year of data is needed.

1.5 Contribution of Study

This project of developing an affordable rain gauge helps in solving a few problems within the industry. First of all, it solves the problem of rain gauge available in the market is too expensive and is not affordable to the common public. The rain gauge is also made to be IoT-connected, reducing human interaction with the prototype for data collection. Data can be obtained through Cloud services, making the process of obtaining the data to be easy. Other than that, the rain rate data obtained from the rain gauge can also be turned into rain fade for analysis of the rain attenuation towards different frequency of signals. It helps the telecommunication industry in making a better solution schemes to deter the attenuation cause by rain. The data is also updated for every minute as compared to the inconsistent updates of rain data in the online source, making the prototype to be even more reliable.

1.6 Outline of the Report

This project report follows the standardised report where it starts with the background of the project with introduction to the study. Then, the problem statement and identification of the problem are stated with the objectives and scope of the project. Chapter 2 covers the preliminary and thorough research of the project. Literature review of articles, conference papers, journals and more has been done to get a deep understanding on the project title and to receive a basic idea on how to start with the project. In the subsequent chapter, the project workflow, methodology in completing the project will be done. This chapter includes the design of the circuit, design of the hardware prototype and the steps and actions taken to complete the hardware prototype before venturing into the collection of the data in the next chapter. In chapter 4, it will be dealing with the raw data collection and short analysis and visualising of the raw data for ease of study and making it more understandable. This chapter will also incorporate the discussion and rationale of the data obtained and also verification and comparison with the data obtained at the nearby rain gauge station to the location of the deployment of the prototype. Lastly, the report reaches it resolution in the final chapter, chapter 5 by providing the concluding statement regarding the research project and also recommendation and improvement for this project's successor with the intention of further developing the prototype to obtain better and more accurate results.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

To fulfill the project outcome, which is to finally develop an affordable rain gauge for rain rate and rain fade measurements, various sources of references such as journals, articles, conference papers, website articles and others have been studied and reviewed before implementing to the project. Topics related to rain gauges, rain rates and rain fades are researched and covered on this study. The following subchapters of this literature review will cover the information and facts related to the project, such as the basics of rain rates and rain fades, rain gauge technologies, application in rainfall measurement and application in rain fade prediction.

2.2 Basics of Rain Rates and Rain Fades

In this subchapter, the basic concepts of rain rates and rain fades will be discussed and explained. The correlation between rain rates and rain fades will also be included in this subchapter.

2.2.1 Basic Concept of Rain Rates

Rain rate or rainfall rate is the measure of the intensity of rainfall by computing the volume of rain that falls over a certain time interval(American Meteorological Society, 2022).

The rain rate is usually given in units of mm/hr (millimeters per hour) or in/hr (inches per hour) (Barani, 2020). The calculation for the rain rate through a rain gauge is as follows:

$$\text{Rain rate (Rainfall rate)}, R = \frac{\text{mm}^3 \text{ of rain}}{\text{mm}^2 \text{ rain gauge opening area } \times \text{ hour}} \tag{2.1}$$

Another equation that can be used to determine the rainfall rate is by (Samad et al., 2021):

$$R = \frac{\text{Rainfall per tip} \times \text{Number of tips}}{\text{Time gap per tip}}$$

(2.2)

According to Barani (2020), using the MANOBS (Manual of Surface Weather Observations), the general categories of intensity of rainfall which is determined by the rain rate are as follows:

Table 2.1: Rainfall intensity and its respective rainfall rate (Barani, 2020)

Rainfall intensity	Rainfall rate
Light rain	$< 2.5\ mm/hr$
Moderate rain	$2.6\ < r < 7.5\ mm/hr$
Heavy rain	$7.6\ < r < 50\ mm/hr$
Violent rain	$> 50\ mm/hr$

As the intensity of the rain increases, the rain fade will also be affected. The rain fade is tied to the rain rate and frequency-related coefficients.

According to the Malaysian website from the Department of Irrigation and Drainage Malaysia, Ministry of Energy Transition and Water Transformation, the categorization of the rainfall intensity or rain rate in one hour is as below:

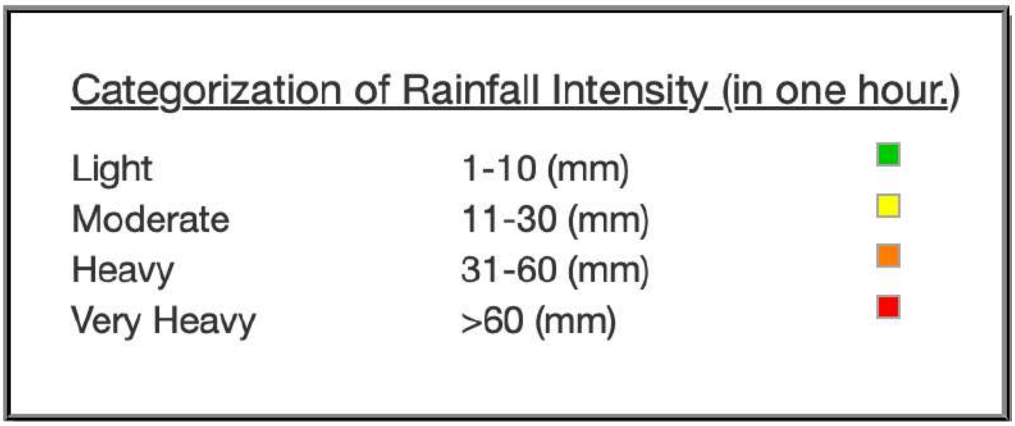

Figure 2.1: Categorization of Rainfall Intensity (in one hour) in Malaysia (Anon, n.d.)

2.2.2 Basic Concept of Rain Fade

The rain fade is also known as rain attenuation. Rain fade depends on the type of rain and the intensity of the rain (Hyde and Bargellini, 2002). The rain fade is also highly affected by the operating frequencies. As the operating frequency exceed 10 GHz, the attenuation of the signal will be very severe, causing disturbance to the transmitted signal (Alozie et al., 2022). The severity of attenuation from heavy rainfall increases for frequencies ranging from 10 GHz to 500 GHz which is from microwave to submillimeter wavelengths (Britannica, n.d.).

To explain on rain fades, when the electromagnetic waves encounter the rain droplets, the waves will be scattered and absorbed, this will cause the strength and quality of the original signal transmitted to be dropped and weakened, causing disturbance to the signal. The relation between rain rate and rain fade can be seen from the equation shown below:

$$Rain\ Fade, \gamma = aR^b \tag{2.3}$$

R represents the rain rate, whereas a and b are the functions of frequencies (Olsen et al., 1978). Hence, to predict the rain fade at a certain location, the rain rate must first be determined at that certain location.

2.3 Theoretical Rain Models

In this subchapter, the different theoretical models of rain models that are used to calculate and predict the rain rate and rain fade will be discussed and explained.

2.3.1 ITU-R P.838 Model

This model is developed by the International Telecommunication Union (ITU). It is used to estimate the rain fade (dB/km). The equation is depicted in the following Equation (2.4):

$$Rain\ Fade, \gamma_R = kR^\alpha \tag{2.4}$$

Where:

γ_R = rain attenuation or rain fade, dB/km

R = rain rate, mm/hr

k = frequency constants depending on the frequency of the link and region's climate

α = frequency constants depending on the frequency of the link and region's climate

The k and α constants can be obtained from the following equations provided by the International Telecommunication Union (ITU). To calculate k and α:

$$log_{10}k = \sum_{j=1}^{4}\left(a_j \exp\left[-\left(\frac{log_{10}f - b_j}{c_j}\right)^2\right]\right) + m_k log_{10}f + c_k \tag{2.5}$$

$$\alpha = \sum_{j=1}^{5}\left(a_j \exp\left[-\left(\frac{log_{10}f - b_j}{c_j}\right)^2\right]\right) + m_\alpha log_{10}f + c_\alpha \tag{2.6}$$

Where:

f = frequency in GHz

k = either k_H (horizontal) or k_V (vertical)

α = either α_H (horizontal) or α_V (vertical)

The value of coefficients for a and b can be obtained from the ITU-R P.838 document (Radiocommunication Bureau, 1992). From the same document, the International Telecommunication Union have also provided the frequency-dependent coefficients of k and α to estimate the rain attenuation at different frequencies ranging from 1 GHz to 1000 GHz. Then, inputting the values to Equation (2.4), we are able to obtain the rain fade in relation to the rain rate. Some examples of the coefficients are shown below:

Table 2.2: Frequency-dependent coefficients for estimating specific rain fade (Radiocommunication Bureau, 1992).

Frequency (GHz)	k_H	α_H	k_V	α_V

10	0.01217	1.2571	0.01129	1.2156
20	0.09164	1.0568	0.09611	0.9847
30	0.2403	0.9485	0.2291	0.9129
40	0.4431	0.8673	0.4274	0.8421
50	0.6600	0.8084	0.6472	0.7871
60	0.8606	0.7656	0.8515	0.7454

The above table shown a small part of the tables provided by the International Telecommunication Union. From the table above, it is observed that as the frequency increases, the value of the coefficients increases. If the rain rate is constant, according to the Equation (2.4), the rain fade or the attenuation of the rain is also increased, causing the quality of the signal transmitted at that specific frequency to be bad and attenuated. This statement fits the theory discussed previously.

2.3.2 Marshall-Palmer Relation

The Marshall-Palmer relation also known as the Z-R relation is used for the estimation of the rain rate based on radar reflectivity factor which is measured from weather radars. The radar reflectivity factor, Z is a measure of the intensity of the radar signal returned from the particles in the atmosphere. The equation of the Z-R relation is shown as below:

$$Z = 200R^{1.6} \tag{2.7}$$

Where:

Z = reflectivity factor

R = rainfall rate

According to Fournier (1999), Z is dependent on the raindrop size distribution and the size of the raindrops (to the sixth power), while R is related to the raindrop size distribution, the size of the raindrops (to the third power) and the fall velocity for a given drop diameter. Fournier (1999) also stated that the Z-R relation is not unique because the rain rates for a certain reflectivity can be different. For tropical regions, such as in Malaysia, the National Weather Service provided the default relationship of the Z-R equation to be different as the Equation (2.7) could not accurately estimate the rain rates in tropical regions. The equation stated above is shown as below:

$$Z = 250R^{1.2}$$

(2.8)

Where:

Z = reflectivity factor

R = rainfall rate

There are also other relationships of the Z-R based on the type of rains. For orographic rain or the rainfall at mountains, the equation is changed to as below:

$$Z = 31R^{1.71}$$

(2.9)

Where:

Z = reflectivity factor

R = rainfall rate

Whereas for thunderstorms, the equation is as below:

$$Z = 486R^{1.38}$$

(2.10)

Where:

Z = reflectivity factor

R = rainfall rate

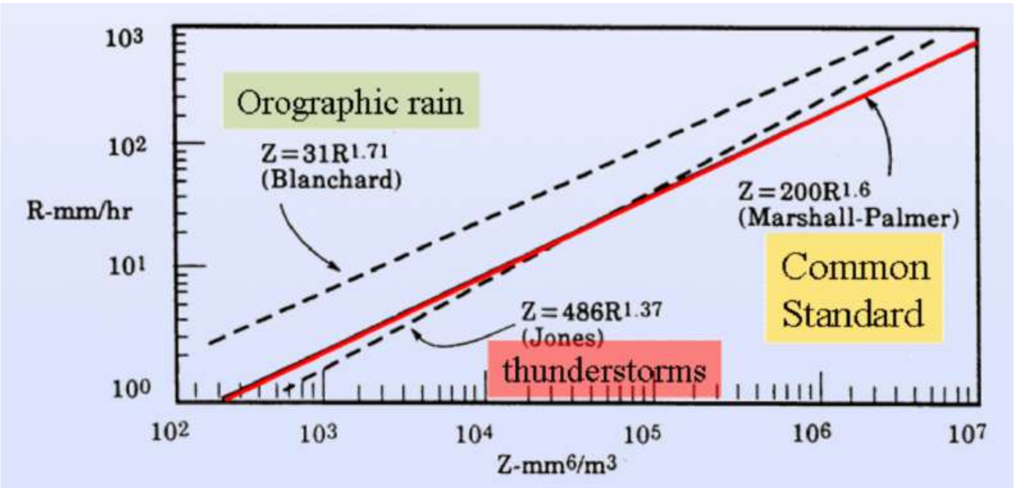

Figure 2.2: Z-R relationship under different conditions of rain (Anon, n.d.)

Hence, it is deduced that as the Marshall-Palmer relation does not have a fixed general equation for a specific location, only when the rainfall rate measurements are done at that location, the rainfall rate will only be more accurate (Anon, n.d.).

2.3.3 Crane Model

The other widely used model for estimating rain fade is the Crane Model. The Crane Model uses the same equation as the ITU-R P.838 model. The more developed model, the Crane Global Model takes the geographical location and climate on a global scale into account while estimating the rain attenuation. It also takes the temperature, relative humidity and the precipitations patterns specific to different regions into consideration while estimating the rain fade.

According to Fashuyi and Afullo (2007), Crane achieved the model by a piecewise representation of the path profile by exponential functions. A satisfactory model is realized by using two exponential functions to cover the range from 0 to 22.5 km, one from 0 to δ(R) km and another one from δ(R) to 22.5 km:

$$A_T(R,d) = \gamma(R)\left(\frac{e^{y\delta(R)}-1}{y} + \frac{e^{zd}-e^{y\delta(R)}}{z}e^{\alpha B}\right)$$

for $\delta(R) < d < 22.5$ km

$$A_T(R,d) = \gamma(R)\left(\frac{e^{y\delta(R)}-1}{y}\right)$$

for $0 < d < \delta(R)$ km

(2.11)

Where:

A_T = horizontal path attenuation (dB)

R = rain rate (mm/hr)

$\gamma(R)$ = specific attenuation = kR^{α} (dB/km)

2.4 Existing Rain Gauge Technologies

In this subchapter, the existing rain gauge technologies and the brief history and development of rain gauge technologies will be explained. The brief introduction to discuss on what is rain gauge used for will also be done.

2.4.1 Brief Introduction about Rain Gauges

A rain gauge is a meteorological measuring device that is used to measure the precipitating rain in a given amount of time per unit area. The basic measuring instrument has a collection container which will be placed in the open area, and the amount of rain per unit time is measured by the heights of the rainwater accumulated in the container per given time and is expressed in terms of millimeters per unit time (Acharya, 2017).

2.4.2 Brief History and Development of Rain Gauge

According to The History of Rain Measurement (n.d.), the first rain gauge was first made by King Sejong, the king of Korea on the fourth month of 1441. The main reason that the king developed this measuring instrument is to help the villagers determine the potential harvest, and tax on the farmers. After around 200 years, on 1661, Christopher Wren invented a rain gauge that uses a tipping bucket that uses the standard of weight and the volume of the liquid precipitation. However, the standards used were hard to determine until Benjamin Franklin in the United States appear (The History of Rain Measurement, n.d.).

2.4.3 Conventional Rain Gauges

The conventional rain gauges include the standard cylindrical rain gauge and the tipping bucket rain gauge. This type of rain gauges is able to provide accurate results but might be costly and may need manual data collection.

2.4.3.1 Cylindrical Rain Gauge

There are two types of cylindrical rain gauge, one is non-recording whereas the other one is recording type cylindrical rain gauge. The example of non-recording rain gauge is the Symon’s Rain Gauge. This device does not record the rain but only collects the rain at a certain location. The measuring of the volume of the rain is done by using an external measuring device, such as the measuring glass (CementConcrete, 2020).

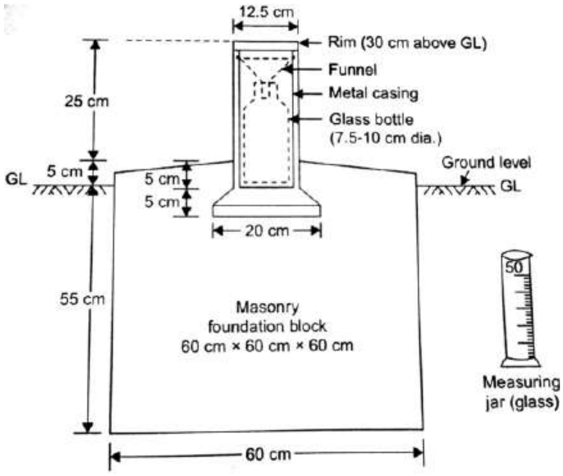

Figure 2.3: Symon's Rain Gauge (CementConcrete, 2020).

The other type of cylindrical rain gauge has the measuring lines on the rain gauge itself. There are no external measuring device needed for this type of rain gauge. The usage and working principle behind the Symon's Rain Gauge are relatively the same but with measuring lines added to the cylindrical container itself.

Figure 2.4: Standard Cylindrical Rain Gauge (Maximum Weather Instruments, 2020)

However, this type of rain gauge requires manual reading of data at intervals of time. The person that deployed the rain gauge must visit the site frequently to record the observation from the rain gauge.

2.4.3.2 Tipping Bucket Rain Gauge

The tipping bucket rain gauge is commonly used in electronic weather stations. This type of rain gauge uses the technology of see-saw where two buckets are mounted on the centre (fulcrum) and balanced using the principle of see-saw. As the rainwater fills the tiny bucket, the part where it is filled with water will be overbalanced and tipped down, touching the screw and activating the small switch that triggers the circuit to record the count. When the tiny bucket tips, the bucket will also be drained and it will return to its original position, allowing another intake of water into the other bucket (Weather Shack, n.d.).

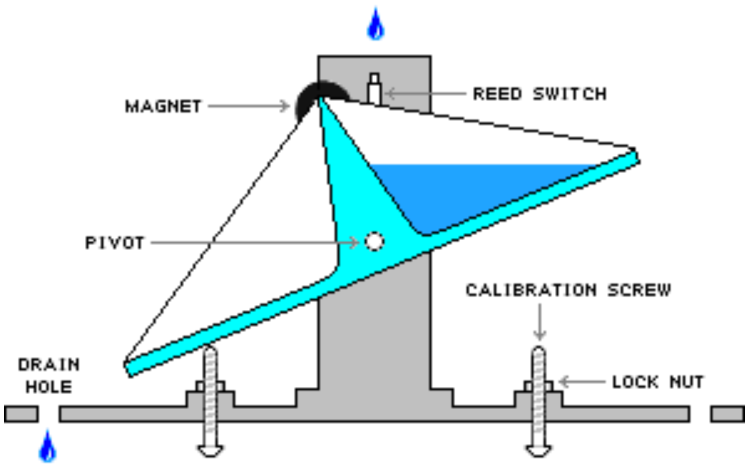

Figure 2.5: Working Principle of Tipping Bucket Rain Gauge (Weather Shack, n.d.)

If the tipping bucket is made wireless, there will be no manual data reading is needed as all the data will be recorded every time the bucket tips.

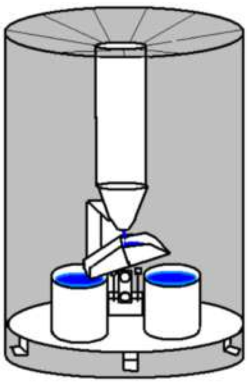

Figure 2.6: Tipping Bucket Rain Gauge (Maximum Weather Instruments, 2020)

2.4.4 Modern Rain Gauges

The modern rain gauges include the use of sensors, radar frequency sensors and many other types of sensors to accurately obtain the data of the rainfall rate. However, these types of rain gauges are usually expensive to be deployed. The modern rain gauges are generally used at weather stations. This type of rain gauge includes impact sensor rain gauge, doppler radar sensor rain gauge, optical sensor rain gauge and more.

2.4.4.1 Impact Sensor Rain Gauge

The impact sensor rain gauge uses an impact sensor which detects the impact of every individual rain drops (Columbia Weather Systems, n.d.). The amount of signal exerting are proportional to the volume of the rain drops, this can then be converted to the rain rate. By using this type of rain gauge, as there are no accumulation of water, flooding, clogging, wetting and evaporation losses can be eliminated (Columbia Weather Systems, n.d.).

Figure 2.7: Orion Weather Station (Columbia Weather Systems, n.d.)

This type of rain gauge is generally used along with weather stations. One of the example of the weather station is the Orion Weather Station. This impact sensor rain gauge also require no maintenance and is highly accurate at measuring the rainfall rate (Scientific Sales, n.d.).

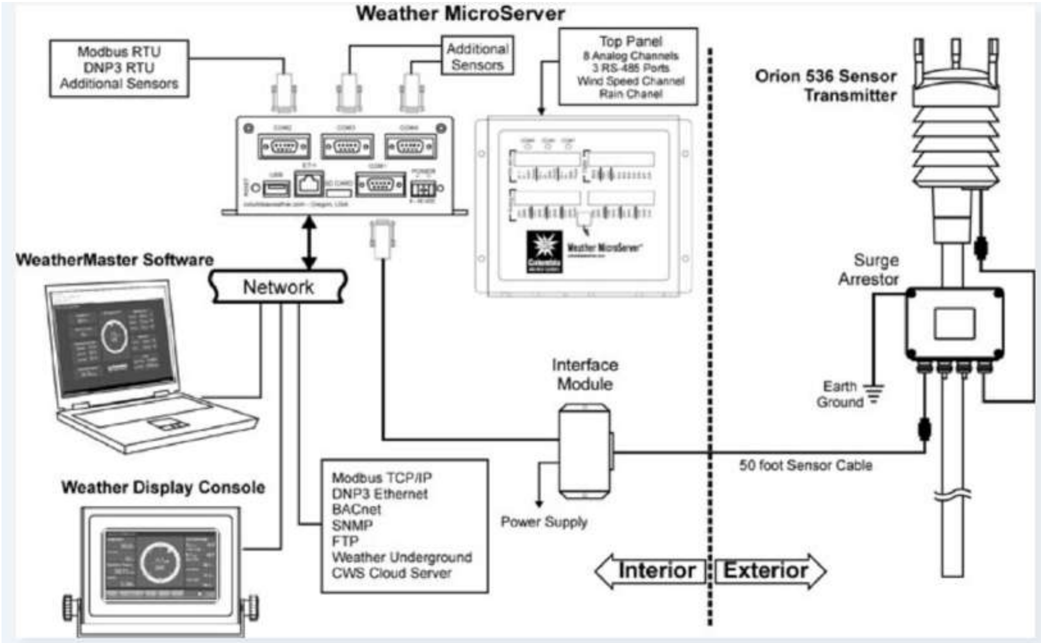

Figure 2.8: Orion Weather Monitoring System Diagram (Columbia Weather Systems, n.d.)

This rain gauge also uses wireless technology which sends the data to an interface to show the rain rate.

2.4.4.2 Optical Sensor Rain Gauge

The optical sensor rain gauge senses the rain drops hitting the outside surface of the sensor using infrared light beams. It measures the parameters according to the number of rain drops and the size of the rain drops (Hydreon, n.d.).

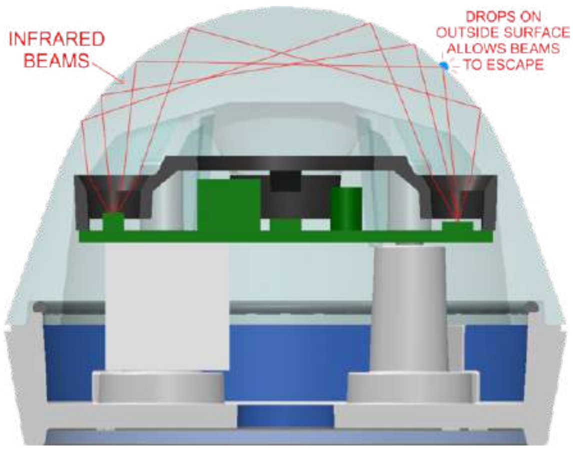

Figure 2.9: Optical Sensor (Hydreon, n.d.)

The optical sensor rain gauge uses algorithms to interpret the data and simulate the output of a tipping bucket rain gauge. However, by using this optical sensor rain gauge, it is able to overcome a lot of cons of using a tipping bucket rain gauge, the sensor do not have funnel which prevents it to clog with leaves and with the dome shape of the sensor, frequent cleaning of the rain gauge is not required.

Figure 2.10: Optical sensor rain gauge (Columbia Weather Systems, n.d.)

However, the accuracy of the data obtained from this sensor may be low and the resolution is also low. This type of rain gauge is best used for mobile and portable applications where exact accurate data is not needed such as for general monitoring of recording rain event and duration (Columbia Weather Systems, n.d.).

2.4.5 Findings and Comparison between Different Rain Gauges Technologies

In summary, there are currently conventional rain gauges and modern rain gauges that uses different sensors. Although the modern rain gauges may have a higher accuracy when measuring the rain rate, but the cost may be very high and contradict the objectives of this project. The comparison between rain gauges technologies can be displayed in the table below:

Table 2.3: Comparison between different rain gauges

Type	Measuring Principle	Accuracy	Cost	Advantages	Disadvantages
Cylindrical	Collects rainwater in a cylinder	Low	Low	1. Simple and low-cost 2. Provides visual measurement	1. Evaporation 2. Require frequent reading and removing the water
Tipping Bucket	Measures rain drops by filling and tipping a bucket	High	Low to Moderate	1. High accuracy 2. Reliable and durable 3. Easy to maintain	1. May overflow during high intensity rain fall
Impact Sensor	Measures force of impact of raindrops	Moderate	Moderate	1. Reliable and robust 2. Suitable for various application	1. Low accuracy during light rainfall 2. Wear and tear mechanical
Optical Sensor	Measure light scatter or absorption of rain drops	High	Moderate to High	1. Non-contact measurement 2. Remote sensing	1. Expensive 2. May be stained and cause low accuracy

From the table shown above, the tipping bucket rain gauge and optical sensor rain gauge has the higher accuracy among all of the other rain gauges. They are more precise than other rain gauges technologies. However, for the optical sensor rain gauge, it is more expensive than the tipping bucket rain gauge, the optical sensor may also be stained and this will cause the accuracy and precision to be low. The optical sensor requires frequent cleaning and maintenance to ensure the data collected to be correct. For tipping bucket rain gauge, the problem is that the container housing the tipping bucket may get overflowed and causing the data not able to collected when it overflows. However, this problem of overflowing may be overcome by putting a larger water drainage hole to ensure all water is flown out after each tip. The difficulty of recreating and developing a tipping bucket rain gauge is also easier than optical sensor rain gauge.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction to the Project Methodology

This chapter will highlight the methods that has been planned and adopted by the author to be used to proceed with the project which is to create an affordable rain gauge to record down and study the rainfall rate for the conversion to rain fade. In this chapter, it will include both planning and execution of the project, on both the hardware and the electronic circuit that are used in the project. The rationale and the principle behind the methods will also be justified at its corresponding parts. The coding for the controlling of the electronic circuit will also be briefly discussed in this chapter.

3.2 Overall Block Diagram

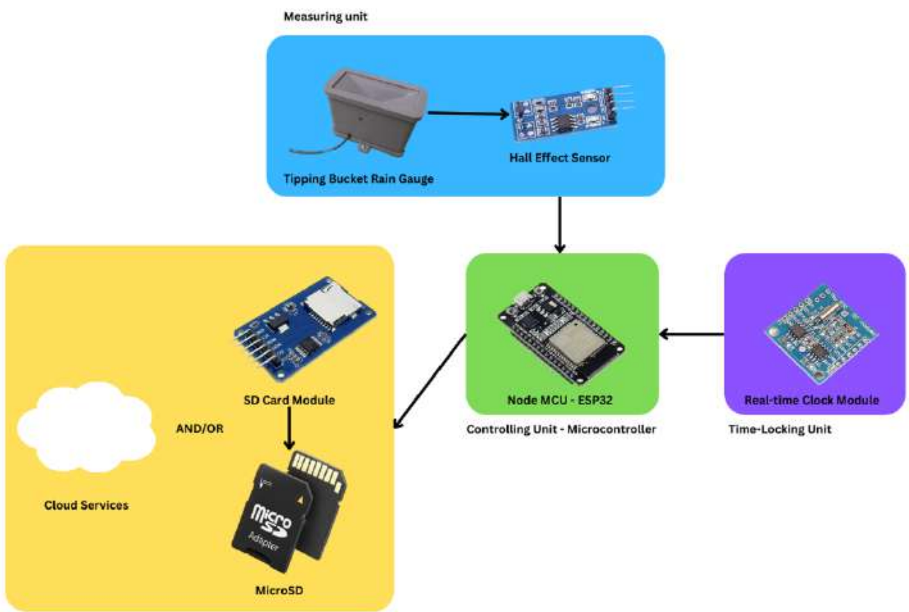

Figure 3.1: Overall block diagram for the prototype

Figure 3.1 above shows the overall block diagram for the whole rain gauge system. The container with the tipping bucket technology and the hall effect sensor will be used as the measuring unit for the rain gauge. Then, the data from the hall effect sensor will be sent to the microcontroller for processing. The time-locking unit which is the RTC module will receive a signal from the ESP32 microcontroller to lock the current time and send it back to the microcontroller for further processing of data. Finally, the data is sent to the microSD module that is connected with a microSD card for storage. At the same time, the data that is saved within the SD card will also be sent to the cloud service. The microSD card will acts as a backup while the cloud will be the main platform where the data will be obtained. The following subchapters will include a more detailed explanation and description of the hardware and software of this project.

3.3 Description of Methods of Approach to the Project

The project of developing an affordable rain gauge prototype is split into two parts, which roughly take up 9 months' time, i.e., two trimesters. In the first part of the project, it includes more of the starting or initiation of the project that includes the research of the project, comparing the components that will be used for the prototype, drafting design for the prototype and also the ordering of parts and electronic components to be used in the project. In the second part of the project, it includes more of the realizing the designed prototype or implementation of the hardware for the prototype. This includes the circuit connection, collection of data, verification of the data and finalizing the product design.

In part one of the final year project, the literature review and research on the project had been done. Resources were obtained through websites, journal articles and as well as resources supplied by supervisor. Further understanding of the project through the discussions with the supervisor were also included along the way during the writing of the report. The research on the concepts used by different rain gauges technologies were done to further understand on the scope of the project. Furthermore, the concept of rain rate, rain fade, theoretical rain models, the way to convert rain rate data to rain fade were also researched and further studied to help in the progress of achieving the objective of this project. With the research done, the tipping bucket rain gauge is chosen to be constructed and further developed. The project is to be done starting from choosing the microcontroller as the brain of the circuit, to control all the other circuits. The other electronic components that will be used alongside the microcontroller will be further described and compared in the next subchapters. A small part of the system design will be done during the first part of the project. However, with the absence of rain, data could not be collected and verified. In the first part of the project, the coding for the microcontroller is also planned to be coded and studied further. In summary, in the first part of the project, research on the project title, further research behind the working principle of rain gauges, and more is done. A simple circuitry for the system is also drafted with part of the coding done.

In Final Year Project Part Two, the simple circuit that only requires the RTC module, the microcontroller, and the hall-effect sensor with tipping bucket rain gauge was tested out and verified. After verifying the simple circuit, the design of the electronic circuit were further developed by adding more function such as storing the data to the cloud and microSD card as backup. During testing out the functionality of the circuit, the code were also developed to match the added function of the rain gauge. The circuit connection and code were fine-tuned along the way. After that, the circuit were soldered onto a donut board as it is more stable than a breadboard before being deployed for field-testing. Then, before the prototype being field-tested, the prototype were calibrated and tested by using syringe and scale measurement method to obtain the actual rain volume obtained for every single tips. After calibration of the prototype, the developed rain gauge were deployed and field-tested. Raw data is collected and obtained from the cloud service and further fine-tuning within the codes are done to obtain better results.

The following subchapters will discuss in detailed and further regarding the process and method that were taken to realize and complete the project.

3.3.1 Project Timeline and Overall Flow of the Project

Gantt Chart

No.	Project Activities	Planned Completion Date	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16
1.	Project discussion with supervisor and research on the project title	2023-06-30	■	■														
2.	Research and literature review on the project	2023-07-28		■	■	■	■	■										
3.	Discussion and prototype design planning	2023-08-04							■	■								
4.	Drafting circuits for the prototype and research on the materials to be used for the prototype	2023-08-18								■	■	■						
5.	Decide on the materials to be used and the components to be bought	2023-08-19										■						
6.	Analysis on the proposed design and materials	2023-08-25											■	■				
7.	Report writing and project presentation 1	2023-09-26												■	■	■		

Figure 3.2: Gantt Chart of the Project (FYP1)

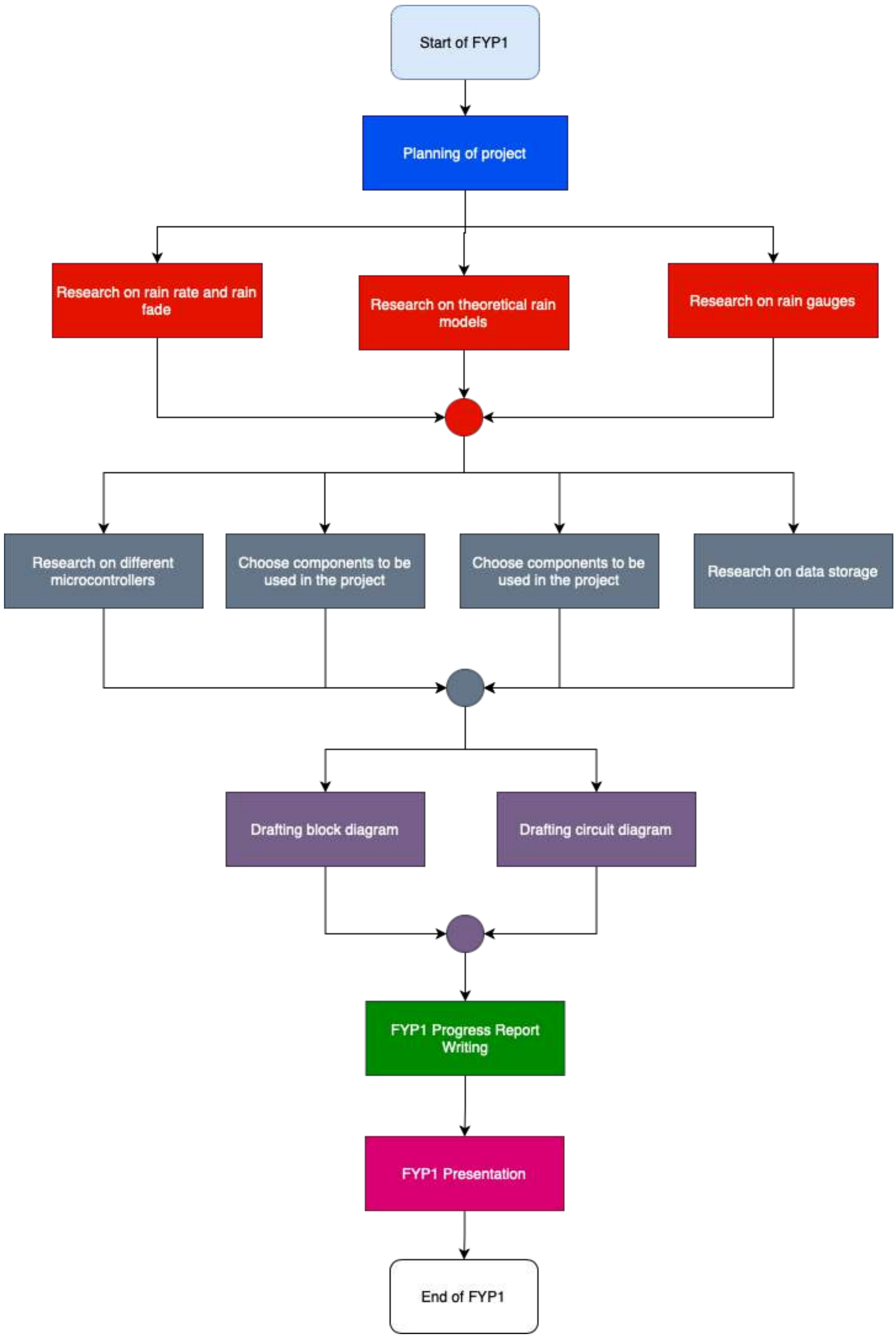

Figure 3.3: Project Flow of FYP1

Gantt Chart

No.	Project Activities	Planned Completion Date	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17
1.	Creating a simple circuit for the prototype and testing it on a breadboard.	2024-02-18	■	■	■														
2.	Develop an algorithm for the microcontroller coding and testing sending rain rate data and relevant information to cloud.	2024-03-03			■	■	■												
3.	Finalising the circuit for the prototype and ensuring every parts of the circuit works well.	2024-03-24						■	■	■									
4.	Soldering the components to donut board.	2024-03-31								■	■								
5.	Deploy and field test the prototype and ensure that it works well and debug any problems with the prototype	2024-04-21										■	■	■					
6.	Continue maintaining the prototype to look for errors.	2024-02-17											■	■					
7.	Report writing and project presentation 2.	2024-05-03												■	■	■			

Figure 3.4: Gantt Chart of the Project (FYP2)

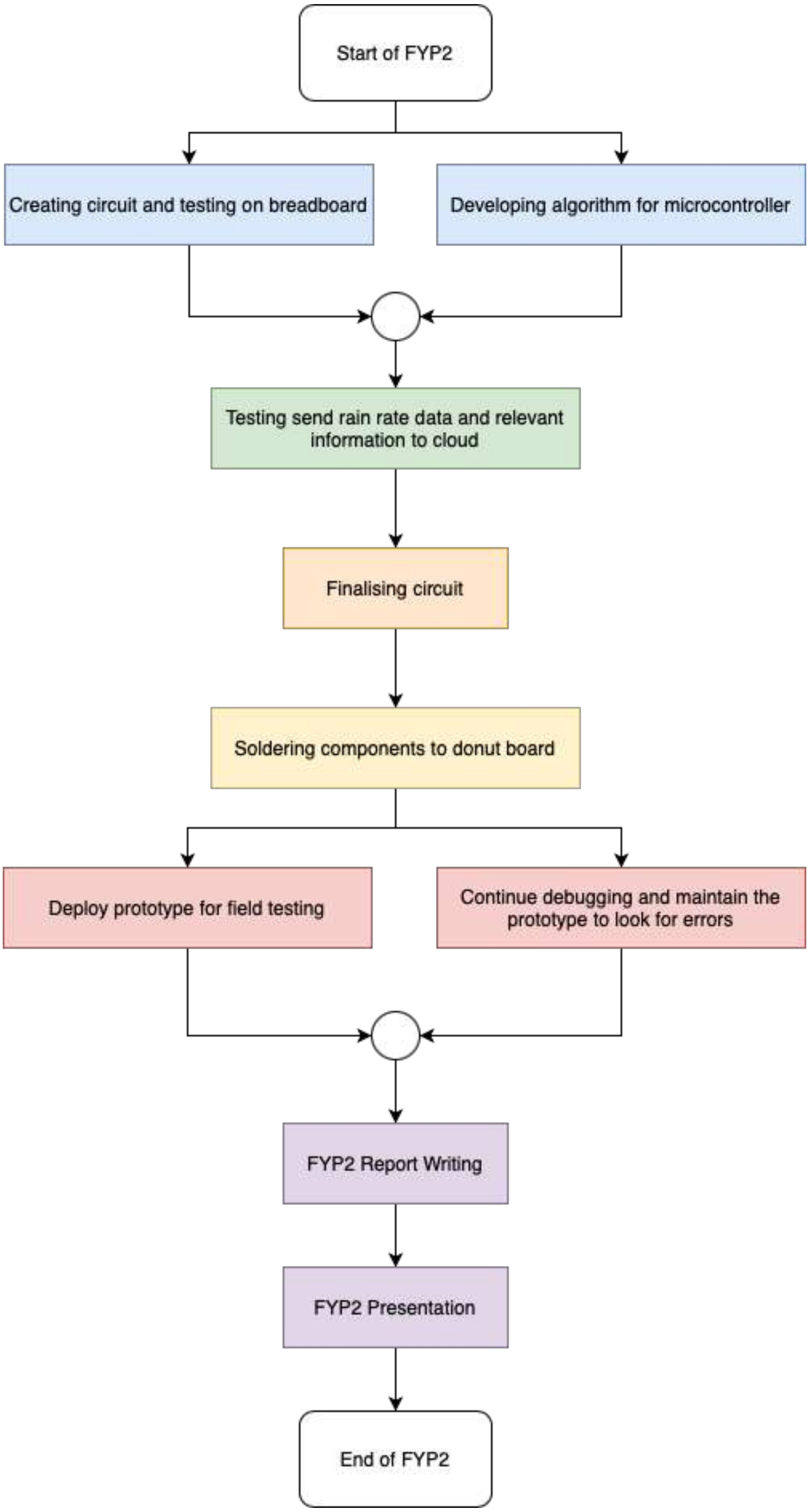

Figure 3.5: Project Flow of FYP2

3.4 Design of the Prototype

In this subchapter, the discussion on the hardware and the components that will be used to form the final prototype will be done.

3.4.1 Hardware Components for the Prototype

The first component that will be discussed is the microcontroller of the circuit, which acts as the center or the brain of the entire circuit. It handles the process of receiving signals, conditioning it to readable data and sending the data to different path for storage both wirelessly and wired.

Table 3.1: Choices of Microcontroller

Item	Microcontroller
NodeMCU ESP32	 Figure 3.6: NodeMCU ESP32
NodeMCU ESP8266	 Figure 3.7: NodeMCU ESP8266
Arduino UNO R3	 Figure 3.8: Arduino UNO R3

Raspberry Pi Model 4	 Figure 3.9: Raspberry Pi Model 4

For this project, four microcontrollers were listed out to be suitable for the project. However, the chosen microcontroller is NodeMCU ESP32. The comparison between the four microcontrollers is listed as below:

Table 3.2: Comparison between the four microcontrollers

Microcontroller	ESP32	ESP8266	Arduino UNO R3	Raspberry Pi 4
CPU Speed	160 MHz	80 MHz	16 MHz	1.5 GHz
RAM	520 KB	160 KB	2 KB	2 GB
Memory	4 MB	4 MB	32 KB	MicroSD card slot
GPIO Pins	36	17	14	40
Wireless Connectivity	Wi-Fi, Bluetooth	Wi-Fi	None	Wi-Fi, Bluetooth
Price	~RM30	~RM15	~RM100	~RM270
Harsh Environment Suitability	Moderate	Moderate	Low	Low

From the table above, although the Raspberry Pi Model 4 is far more superior and has more functionality than all the other listed microcontrollers, but the ESP32 is still chosen due to the high pricing of the Raspberry Pi Model 4. All the functions within the Raspberry Pi may also not be used for the project, making the wastage of resources. According to Teja (2021), Raspberry Pi is more often used for complicated projects. Next, comparing ESP32, ESP8266 and Arduino UNO R3 microcontrollers, ESP32 is the most suitable choice to be used to develop the rain gauge, this is because it has Wi-Fi and Bluetooth built in, more GPIO pins than both the microcontrollers, and higher RAM and Memory than the ESP8266 and Arduino UNO R3 microcontrollers. Comparing ESP32 and Arduino UNO R3, ESP32 is significantly more powerful than the Arduino, this allows ESP32 to be able to handle more tasks (Anon, n.d.). Other than that, ESP32 can also be programmed using the Arduino IDE (Anon, n.d.). The library from Arduino is also usable by ESP32. This makes ESP32 to be versatile and has more flexibility.

After that, the next components that will be discussed is the sensor that will be used to detect the tips from the tipping bucket rain gauge. The sensor that will be used in the project is a Hall Effect Sensor which detects magnetic field. In this rain gauge project, the A3144 Hall Effect Sensor is chosen. According to (Tessie, 2021), the A3144 Hall Effect Sensor is similar to a switch which will change states from High to Low or Low to High but is more sensitive to magnetic flux and it is ideal for severe environments. For the rain gauge, it will be deployed under different weather conditions where it will be exposed to the Sun and also the rain, making the A3144 Hall Effect Sensor to be suitable for this project.

Figure 3.10: A3144 Hall Effect Sensor

The sensor will be placed closed to the tipping bucket that has a magnet attached to it to detect the tips from the rain gauge and record the data more accurately.

Next, a real-time clock module will also be used in this project. A real-time clock is used in a project to record the current date and time (Arathy, 2021). The chosen real-time clock module chosen to be used for this project is the DS1307 RTC.

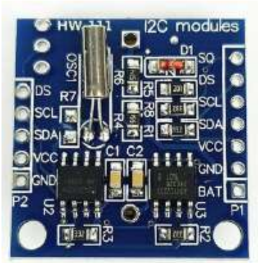

Figure 3.11: RTC DS1307 Module

With the use of the real-time clock module, at the events when rain occurs, the DS1307 will be able to timestamp the data accurately. Then, datalogging of rainfall data will also be able to be logged more accurately with the date and time logged together. This RTC module will be used to lock the date and time with respect to the time that the data is obtained from the tipping-bucket rain gauge.

3.4.2 Housing Material of Tipping Bucket Rain Gauge

The chosen material to be used for the construction of the tipping bucket rain gauge is 3D printing material. There were several materials that can be used to design this rain gauge, among them includes aluminium, stainless steel and 3D printing material. The list were short-listed to stainless steel and 3D printing material. Most of the tipping bucket rain gauges that are available in the market use stainless steel to make the rain gauge (Anon, n.d.). However, the usage of stainless steel to proceed with the construction of the prototype for our project may not be suitable. The below table displays the comparison between stainless steel and 3D printing material as the housing for the tipping bucket rain gauge:

Table 3.3: Comparison between Stainless Steel and 3D Printing Material

Property	Stainless Steel	3D Printed
Durability and Longevity	High durable and long-lasting	Moderate durability
Accuracy	Accurate	Accurate, but may need calibration
Customization	Limited	Easily customizable
Variety	Only stainless steel	Many options:

		- PLA - PLA+ - PETG - ABS
Initial cost	High	Low
Production speed	Slow	Fast

According to the table above, it can be seen that stainless steel may be durable and can last for a very long time, however, the number of advantages of using a 3D printed rain gauge exceed the use of stainless steel. The 3D printed rain gauge is also easily customizable and has many options in material. The costing also plays a major role for this choice of material. For prototyping purposes and further development of the product, 3D printed material is deemed to be more superior than the use of stainless steel.

Within the 3D printing material, there are few other options such as the PLA (Polylactic Acid), PLA+ (Polyactictic Acid Plus), PETG (Polyethylene Terephthalate Glycol), ABS (Acrylonitrile Butadiene Styrene) and more. The final chosen material is PLA+. The comparison between the different 3D printing material can be seen below:

Table 3.4: Comparison between different types of 3D printing material

Property	ABS	PLA	PLA+	PETG
Durability	High	Moderate	Moderate to High	Moderate to High
Difficulty in Printing	Hard	Easy	Easy	Moderate
Heated Printing	Yes	No	No	No
Environmental Benefits	No	Biodegradeble	Biodegradeble	No
Cost	Moderate to High	Low to Moderate	Low to Moderate	Moderate

The reason why PLA+ is chosen as the housing material for the tipping bucket rain gauge is because it is balanced in all aspects, it is durable, easy to be used and cost-effective. For this project, the cost is a big factor when it comes to deciding materials to be used.

3.4.3 Software for the System

The software that is used in this project is only the IDE software used for coding, verifying and uploading the code to the ESP32 microcontroller, the cloud service software and a circuit electronics design software.

The IDE software used in this project is the Arduino IDE. Arduino IDE uses a variant of C++ programming language with addition of special methods and functions and is used generally with Arduino. However, this IDE is also used along with ESP32 as they are compatible and the board information of ESP32 can be downloaded within the software. The software is used to debug the codes, verify the codes and upload the codes to the ESP32 microcontroller through the use of an USB cable to control the whole system.

Figure 3.12: Arduino IDE logo

Next, the other software/website being used is the Google Sheets by Google. Google Sheets is used to store all of the data being generated and processed by the ESP32 controller. The data is sent to Google Sheets by means of wireless communication and API authentication.

Figure 3.13: Google Sheets Logo

The next software being used in this project is the circuit electronics design software which the circuit diagram is being drafted and finalised before actually proceeding to the hardware part for the project. The software used is the Fritzing software.

Figure 3.14: Fritzing software logo

3.4.4 Data Storage

There are two ways that the data from the rain gauge can be stored in this project. One is to store the data into a MicroSD card and manually obtaining the data at certain times and another one is to send to the cloud for storage.

In this project, the method that is used is to send the data to the cloud for storage. According to Kent (n.d.), cloud storage is better than local storage (MicroSD card) as MicroSD card can cause data loss as it can be easily damaged, malfunctions or breaks. Moreover, MicroSD card also requires a person to manually obtain the data every time while cloud storage allows a person to access the data at anywhere with internet connection. However, even if MicroSD card may not be that reliable to be used as the main data storage method, but it still can be serves as a backup for the storage of data.

For the cloud services, the chosen service is Google Sheets where the data will be updated to the cloud at every minute to the cloud automatically. The data that is uploaded to the cloud will be in the form of rain rate per minute and can be easily accessed with a Google account.

3.5 Electronic Circuit Design and Software Programming

In this subchapter, the detailed circuit design, detailed description of components, working principle behind the circuit and the relation between the coding programming and electronics will be explained and discussed.

3.5.1 Overall Electronic Circuit Design

The design schematic circuit diagram for the whole system is shown below. The schematic design shows the measuring unit, which is the A3144 Hall Effect Sensor, the time-locking unit, RTC DS1307 module, controlling unit, ESP32 microcontroller and data storage unit, the MicroSD card module.

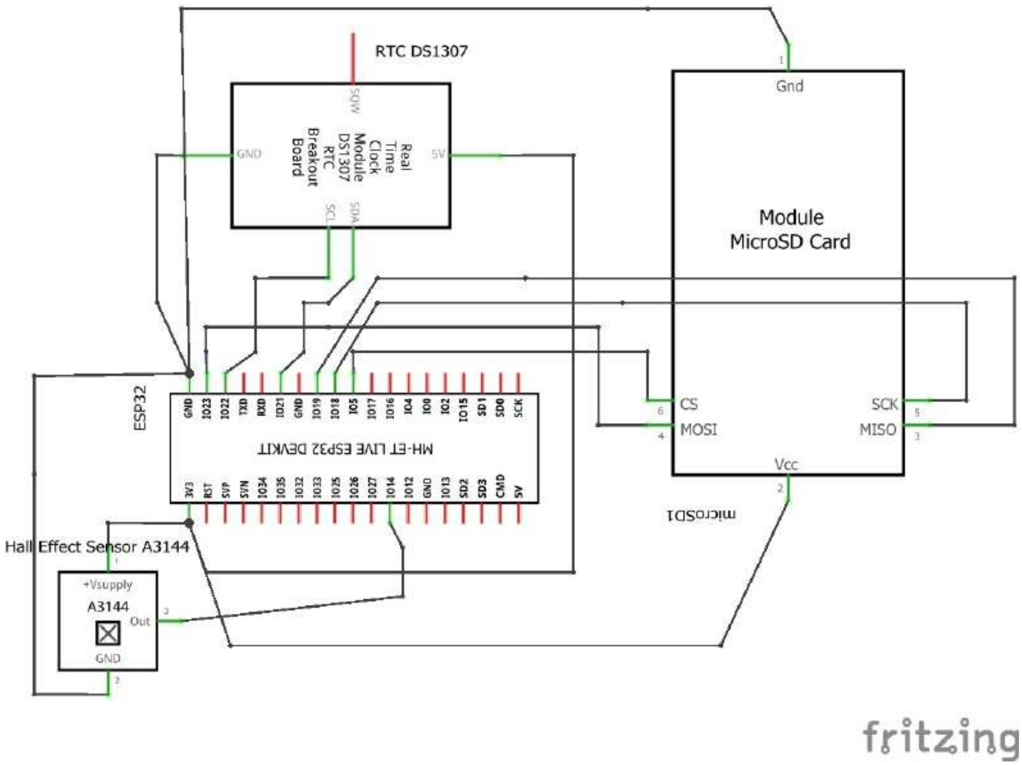

Figure 3.15: Schematic Circuit Diagram for the whole system

Table 3.5: Pins used by each components

Component	Pins used
ESP32	3V3 pin
	GND pin
	GPIO14
	GPIO5
	GPIO18
	GPIO19
	GPIO21
	GPIO22
	GPIO23
A3144 Hall Effect Sensor	Vsupply pin

	Data Out pin
	GND pin
RTC DS1307 module	5V pin
	SDA pin
	SCL pin
	GND pin
Mini MicroSD Card Module	Vcc
	MOSI
	MISO
	SCK
	CS
	GND pin

Table 3.5 shows the pins that are used by each of the components within the circuit. The detailed description of the pins and the pin connections will be discussed further in the next subchapters.

3.5.2 Detailed Description of Components used in the System

3.5.2.1 ESP32 Microcontroller

The ESP32 used in this project is the ESP32-WROOM-32 model which has 30 pins in total, with each of the pins having its own special functions. There are two power pins, VIN and 3V3. In this project, only 3V3 pin is used to power up external components from its on-board voltage regulator. The 3V3 pin is connected to every component within the system, the RTC module, MicroSD card module and also the Hall Effect sensor. The GPIO pins of ESP32 is configured to work with each of the components based on the function of the components.

GPIO14 of ESP32 in this project is configured to be the input pin for recording the tipping count from the Hall Effect sensor. Whenever it receives a signal from the Hall Effect sensor, the input will register as one tip.

For the connection between the RTC module to the ESP32, GPIO22 and GPIO21 is connected to the SCL and SDA pins of the RTC module respectively. For ESP32, it does not have a dedicated pins for I2C pins but it allows for flexible pin assignment, meaning that any of the pins can be configured to be I2C pins, SCL (clock line) and SDA (data line). However, for convenience, the default I2C pins commonly used are the GPIO21 (SDA) and GPIO22 (SCL) as shown in Figure 3.16 (Anon, n.d.).

Furthermore the connection between the MicroSD card module and ESP32 module includes GPIO5, GPIO18, GPIO19 and GPIO23 of the ESP32. These pins are the SPI pins or the Serial Peripheral Interface pins that are used for sending and receiving data between a master (controller) and a slave (peripheral) (Anon, n.d.). By default, GPIO5 pin is the CS pin or Chip Select pin which is used to select the device when multiple peripherals are used on the same SPI bus, GPIO18 represents the SCK pin or Serial Clock pin, GPIO19 is the MISO pin or Master In Slave Out pin and finally GPIO23 as the MOSI pin or Master Out Serial In pin (Anon, n.d.). Each of the pins connected to the respective pin on the MicroSD card module pins. The figure below shows the pinout diagram of the ESP32-WROOM-32 model:

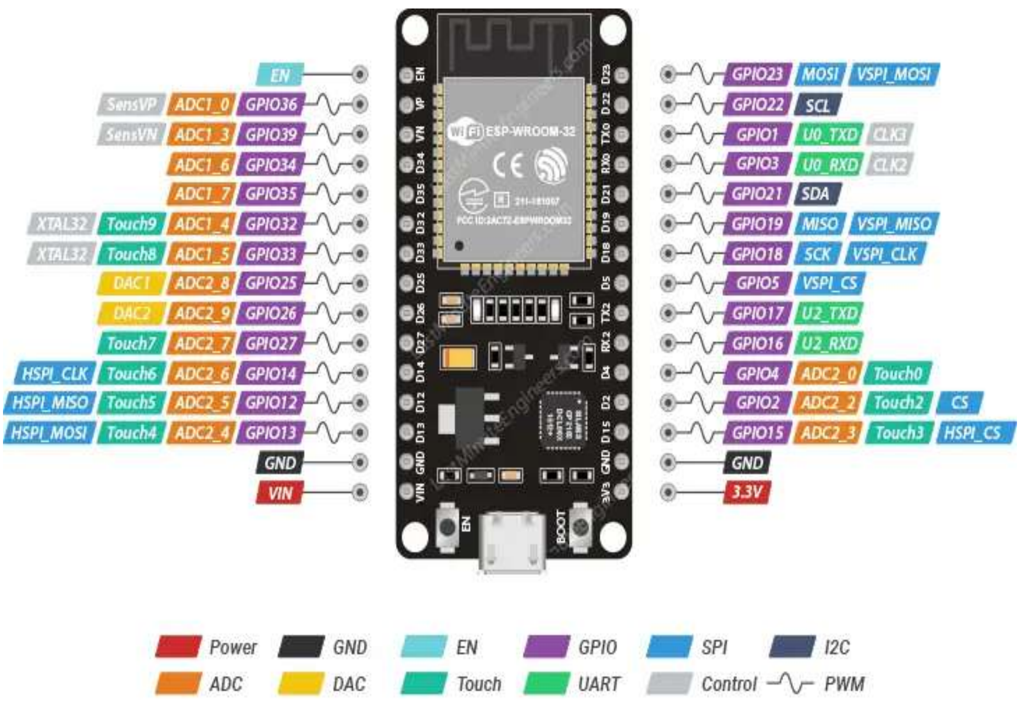

Figure 3.16: Pinout of ESP32-WROOM-32

Table 3.6: Pin Connection of ESP32 in this Project

Pin No.	Pin Name	Pin Description	Connection

30	3V3	Providing 3.3V power to external components	Connected to Vsupply of Hall Effect Sensor A3144, 5V pin of RTC DS1307 module and Vcc of MicroSD module
14 and 29	GND	Ground pin of ESP32	Connected to common ground of the whole system
11	GPIO14	General purpose input/output pin	Connected to Data Out pin of Hall Effect Sensor A3144
23	GPIO5	General purpose input/output pin or default Chip Select pin (CS)	Connected to CS pin of MicroSD module
22	GPIO18	General purpose input/output pin or default Serial Clock Pin (SCK)	Connected to SCK pin of MicroSD module
21	GPIO19	General purpose input/output pin or default Master In Slave Out pin (MISO)	Connected to MISO pin of MicroSD module
20	GPIO21	General purpose input/output pin or default SDA pin	Connected to SDA pin of RTC DS1307 module
17	GPIO22	General purpose input/output pin or default SCL pin	Connected to SCL pin of RTC DS1307 module
16	GPIO23	General purpose input/output pin or default Master Out Slave In pin (MOSI)	Connected to the MOSI pin of MicroSD module

Note that the numbering for Pin No. in this table refers to the number counted from left side top to bottom then right side left to bottom.

3.5.2.2 Hall Effect Sensor A3144

In this project, the tipping bucket is fixed with a Hall Effect Sensor by its side. The Hall Effect Sensor is the sensor that is responsible in detecting the tips from the tipping bucket using the principle of magnetic field. The model of Hall Effect Sensor used in this project is the A3144, a type of Hall Effect Sensor that gives digital output rather than an analog output. When the sensor detects a magnet, the output will go low, and if not, it remains high all the time (Anon, 2022). Hence, by knowing this characteristics, the coding within the ESP32 is set to detect every low signal from the Hall Effect Sensor.

The Hall Effect Sensor has an on-chip Hall voltage generator to sense magnets, a comparator that amplifies the voltage, and an Schmitt trigger that provides switching hysteresis to reject noise, and open-collector output. Whenever a magnetic flux density around the Hall Effect Sensor exceeds the threshold, the output is turned on (low output), and this state is held until the magnetic flux density falls below the threshold, making the output state to off (high output) (Anon, 2022).

In this project, a magnet is attached to the tipping bucket, and a Hall Effect Sensor is attached to the side of the rain gauge wall. Whenever the tipping bucket tips, the Hall Effect Sensor will turn on and a signal will be sent to the ESP32 to record the tipping count. The figures below show the integration of Hall Effect Sensor on the rain gauge body and the attachment of the magnet on the tipping bucket:

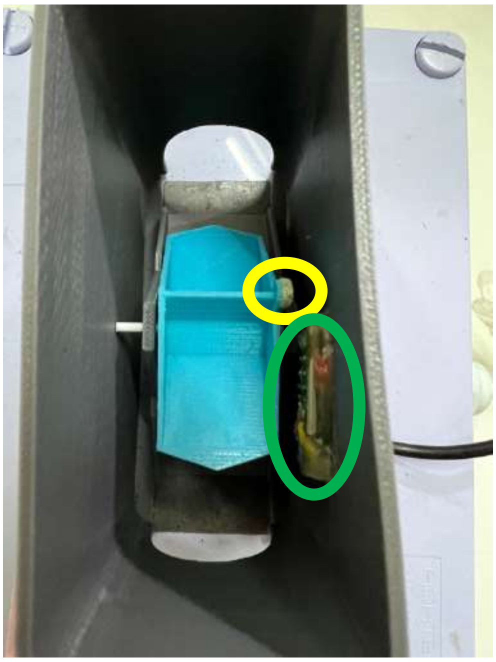

Figure 3.17: Image of the rain gauge prototype (top view)

The yellow circle indicates the magnet that is attached to the tipping bucket and the green circle shows the Hall Effect Sensor A3144.

Figure 3.18: Image showing when the Hall Effect Sensor senses the magnet

The above figure shows the position where the Hall Effect Sensor will sense the magnet. At this point, the magnet is closed to the Hall Effect Sensor, triggering the sensor, providing an ON signal to the ESP32 and represents a tip from the tipping bucket.

Table 3.7: Pin Connection of Hall Effect Sensor A3144 in this Project

Pin No.	Pin Name	Pin Description	Connection
1	Vsupply	Input power pin	Connected to 3V3 pin of ESP32
2	Ground	Ground pin	Connected to the common ground of the whole system
3	Data Out Pin	Output pin	Connected to GPIO14 pin of ESP32

3.5.2.3 RTC DS1307 Module

A real-time clock module is used in this project. The real-time clock module is a module that keeps track of the date and time even when a supply is removed. It locks the date and time when it was first connected and adjusted by a line of coding that was uploaded to the ESP32. The RTC also has a battery that allows it to operate when a voltage supply was removed from the circuit (Meaney, n.d.). The RTC module used in this project is the RTC DS1307 module.

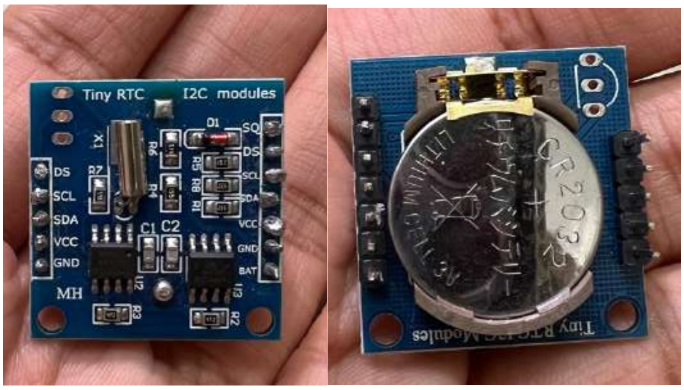

Figure 3.19: Front and Back view of RTC DS1307 Module

Table 3.8: Pin Connection of RTC DS1307 Module in this Project

Pin Name	Pin Description	Connection
SCL	Communicates with other devices, digital I/O which transmits and receives data	Connected to SCL pin of ESP32

SDA	Communicates with other devices, digital input which locks data on SDA	Connected to SDA pin of ESP32
Vcc	Power Supply Pin	Connected to 3V3 pin of ESP32
GND	Ground Pin	Connected to the common ground of the whole system

3.5.2.4 MicroSD Card Module

This module is purely used to receive the processed data from the ESP32 and stores it to the MicroSD card that is connected to the module. The MicroSD card module used in this project is the mini module, which is the 3V3 module and only requires 3.3 V to function.

Figure 3.20: Front and Back view of MicroSD card Module

Table 3.9: Pin Connection of MicroSD Card Module in this Project

Pin No.	**Pin Name**	**Pin Description**	**Connection**
1	3V3	Power supply pin. Input 3.3 V.	Connected to 3V3 pin of ESP32
2	CS	Chip Select pin.	Connected to GPIO5 of ESP32 (CS pin)
3	MOSI	Master Out Slave In pin.	Connected to GPIO23 of ESP32 (MOSI pin)
4	CLK	Serial Clock pin	Connected to GPIO18 of ESP32 (SCK pin)

5	MISO	Master In Slave Out pin	Connected to GPIO19 of ESP32 (MISO pin)
6	GND	Ground pin	Connected to the common ground of the whole system

3.5.3 Calibration of Tipping Bucket Rain Gauge

Before deploying the tipping bucket rain gauge for data collection, measurements have to be made to ensure how many mm of rain every tips of the tipping bucket represents. In this project, a method to test how many mm does the tipping bucket represent per tip is developed. A syringe with accurate measurement is first placed above the rain gauge, then water is slowly dripped onto the rain gauge, simulating artificial rain.

In this case, the tipping bucket is found to be tipping every 1.5ml of water is being dripped onto the rain gauge. Then, the rain gauge opening area is measured to be 28.9 cm^2. Using the basis of equation (2.1, it can be calculated as below:

$$mm\ per\ tip = \frac{mm^3\ of\ water}{mm^2\ of\ opening\ area}$$

$$As\ known, 1\ ml = 1000\ mm^3, hence\ 1.5\ ml = 1500\ mm^3$$

$$Then, 28.9\ cm^2 = 2890\ mm^2$$

$$mm\ per\ tip = \frac{1500}{2890} = 0.519\ mm$$

Hence, it is found out that the tipping bucket tips at every 0.519 mm of rain. Next, to verify the tipping bucket tips at every 0.519 mm, another method which is by using the scale is done. A weighing scale that is tared when the whole prototype is placed onto the scale. 15 ml of water is poured slowly into the rain gauge, then the number of tips is recorded. From this method, the number of tips recorded are 10 tips. Then using the calculation below, the mm per tip can be obtained:

$$ml\ per\ tip = \frac{15ml}{10} = 1.5\ ml$$

$$mm\ per\ tip = \frac{1500}{2890} = 0.519\ mm$$

Therefore, it is verified that every tips from the tipping bucket represents 0.519 mm. However, for standardisation purpose, the tip is said to be 0.5 mm per tip.

Figure 3.21: Measurement of Rain Gauge Opening Area

3.5.4 Data Collection Method (Software)

3.5.4.1 Cloud Service and MicroSD

As discussed earlier, the storage of data for this project will be in two modes. One is to be saved in a physical MicroSD card as a backup and the other is to send to Google Sheets for storage. The process of storing the data to the cloud is automatic. This is achieved by the help of Google Apps Scripts, which is a service provided by Google that provides a development platform to work with different applications by Google which includes the Google Sheets.

To explain this, the Google Apps Scripts project is deployed and the deployment ID is added into the Arduino IDE coding. This provides a linkage between the Google App Scripts and the Arduino IDE coding. Then within the Google Apps Scripts, it is always posting HTTP GET requests to receive parameters to be added into the Sheets. Within Arduino IDE, a function is added to send the data received from the rain gauge to be appended to the Google Sheets. The Arduino code constructs a URL by using the Google App Script ID and the parameters and then sends an HTTP GET request to the URL using the "HTTPClient" object. The GET requests from the code triggers the execution of the function within the Google App Script that, passing the parameters in the

URL and then the Google App Script that was set in the beginning will process the parameters, then append to a new row of data to the Google Sheets that is specified within the Google App Script code. The data saved in Google Sheets are the Date, Time, Rain rate per minute and Weather (Accumulated).

For the MicroSD card, a file is opened in txt format and data is appended to the SD card every minute. The data saved within the SD card is exactly the same as the data being saved in the Google Sheets as the SD card only provides a layer of backup for the data whenever the data is not uploaded to the Google Sheets.

3.5.4.2 Data Collection Procedure

In this project, data is collected as soon as the ESP32 is powered up. The script file or coding file that is uploaded to the ESP32 is instructed to run continuously without going into sleep mode. ESP32 has a function to be configured to sleep mode at certain specified time, however, in this project, this function is not used. The rain gauge is being operated for 24 hours every day and the data is being uploaded for every 1 minute.

The code in ESP32 is divided into many different functions to detect tipping from the rain gauge (Hall Effect Sensor), to process the tips into readable data and to send the data to both cloud and SD card for storage. This process is looping continuously. The process flow of the data collection procedure can be represented in the flowchart as below:

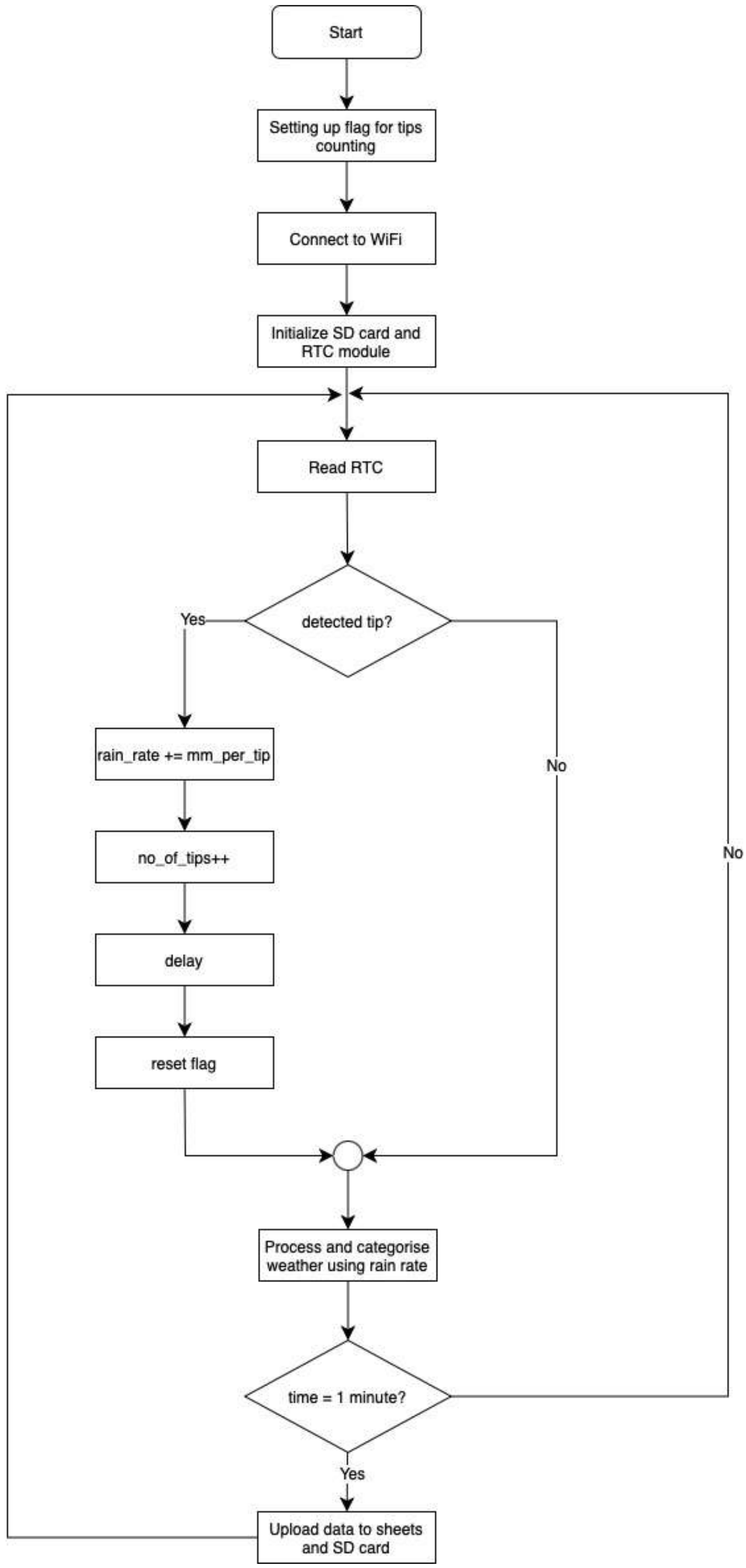

Figure 3.22: Flow of Data Collection Procedure

The process of obtaining the rain rate per minute and uploading to the Google Sheets and SD card is an infinite loop that will continue indefinitely unless the power supply is turned off to stop the data collection process.

3.5.4.3 Data Storage Calculation

Another thing that should be taken into consideration is the storage size for the data. The storage must be big enough to be able to store sufficient amount of data for the research studies, hence, the calculation of storage size is very important. For the storage size limit of Google Sheets, it is allowable to 40,000 rows where only one row is needed for every input of data (Pachori, 2022). The calculation for how long the Google Sheets could last is as below:

$$Time\ (in\ hours)\ for\ the\ data\ to\ reach\ 40{,}000\ data = \frac{400{,}000}{60}$$

$$= 666.67\ hours$$

$$Time\ (in\ days) = 27.78 \approx 27\ days$$

However, it is also to note that Google Sheets allow 200 sheets within one workbook, hence the limit is:

$$Time\ to\ reach\ Google\ Sheets\ limit\ (in\ years) = \frac{27 \times 200}{365}$$

$$= 14.79\ years$$

In the ideal case where every rain rate data is updated to the cloud at every single minute without missing a single data, the total time taken to deplete the given storage from Google Sheets alone is 14.79 years, which is totally sufficient for this research study.

Next, the other mode of data storage used in this project is MicroSD card as a backup. Whenever a single line of rain rate data is missed out to be uploaded to the cloud, it could be found manually in the MicroSD card. For the MicroSD card, one thing to consider is also the storage size and how long would the MicroSD card last for datalogging for every minute. In this project, the MicroSD card size used is 32 GB and a line of text added into a single text file is only 70 bytes. Hence, the calculation for how long the MicroSD card can store the data for is as below:

$$One\ line\ of\ text = 70\ bytes = 0.07\ kB$$

$$1GB = 1{,}000{,}000\ kB$$

$$Total\ no.of\ line\ of\ text\ that\ can\ be\ uploaded = 457{,}142{,}857.1\ lines$$

$$Total\ time\ till\ 32\ GB = 457{,}142{,}857.1\ minutes = 869.75\ years$$

As seen above, it would take more than 800 years until the storage of MicroSD card is all used up, hence it is totally enough for this project research.

3.5.5 Finalised Circuit for the Prototype

The circuit were tested on the breadboard, but the finalised circuit for the prototype is soldered onto a protoboard or more commonly known as a donut board which provides a more stable connection between the connection being used in the circuit. All the components for the circuit is soldered onto a single 12 x 18 cm protoboard. The finalised components being used in the circuit is a ESP32 microcontroller which acts as the brain of the circuit, a Hall Effect Sensor A3144 which acts as the measuring unit of the circuit, a real-time clock module RTC DS1307 that locks the date and time for more accurate timestamp and finally a MicroSD card module that backups and acts as a storage for the rain rate data.

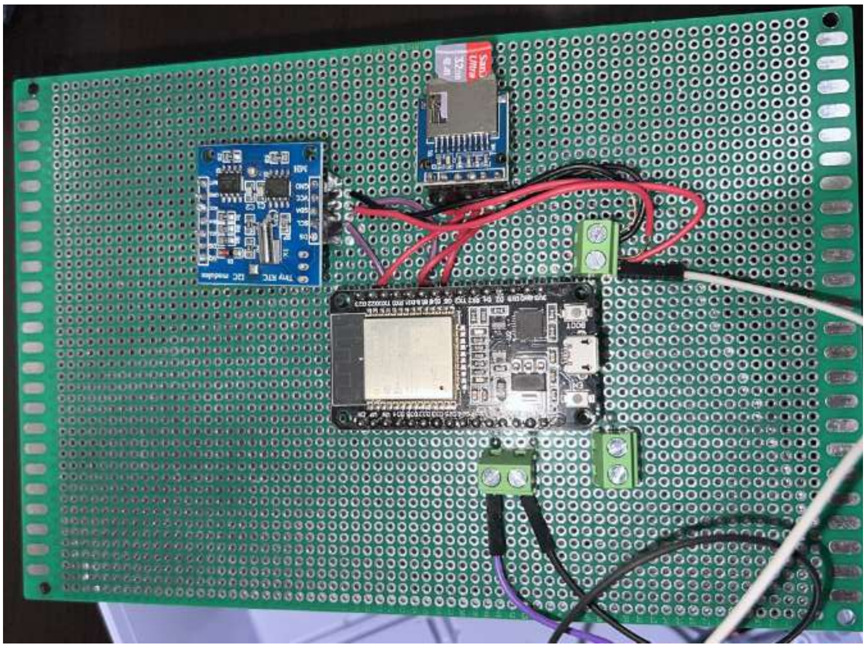

Figure 3.23: Finalised Circuit for Prototype (Front)

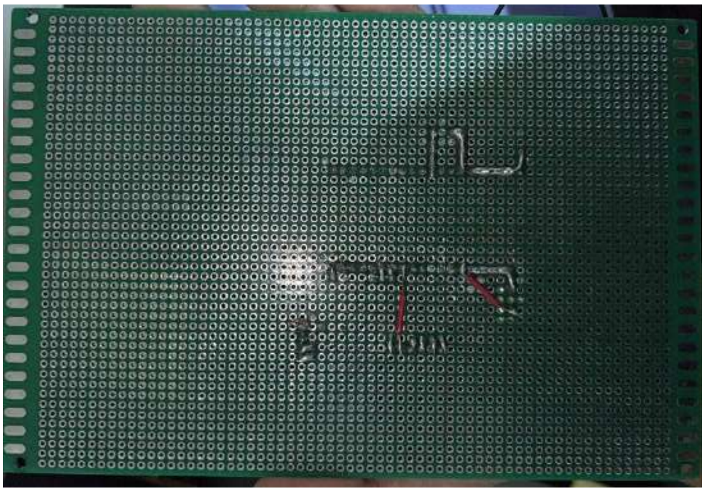

Figure 3.24: Finalised Circuit for Prototype (Back)

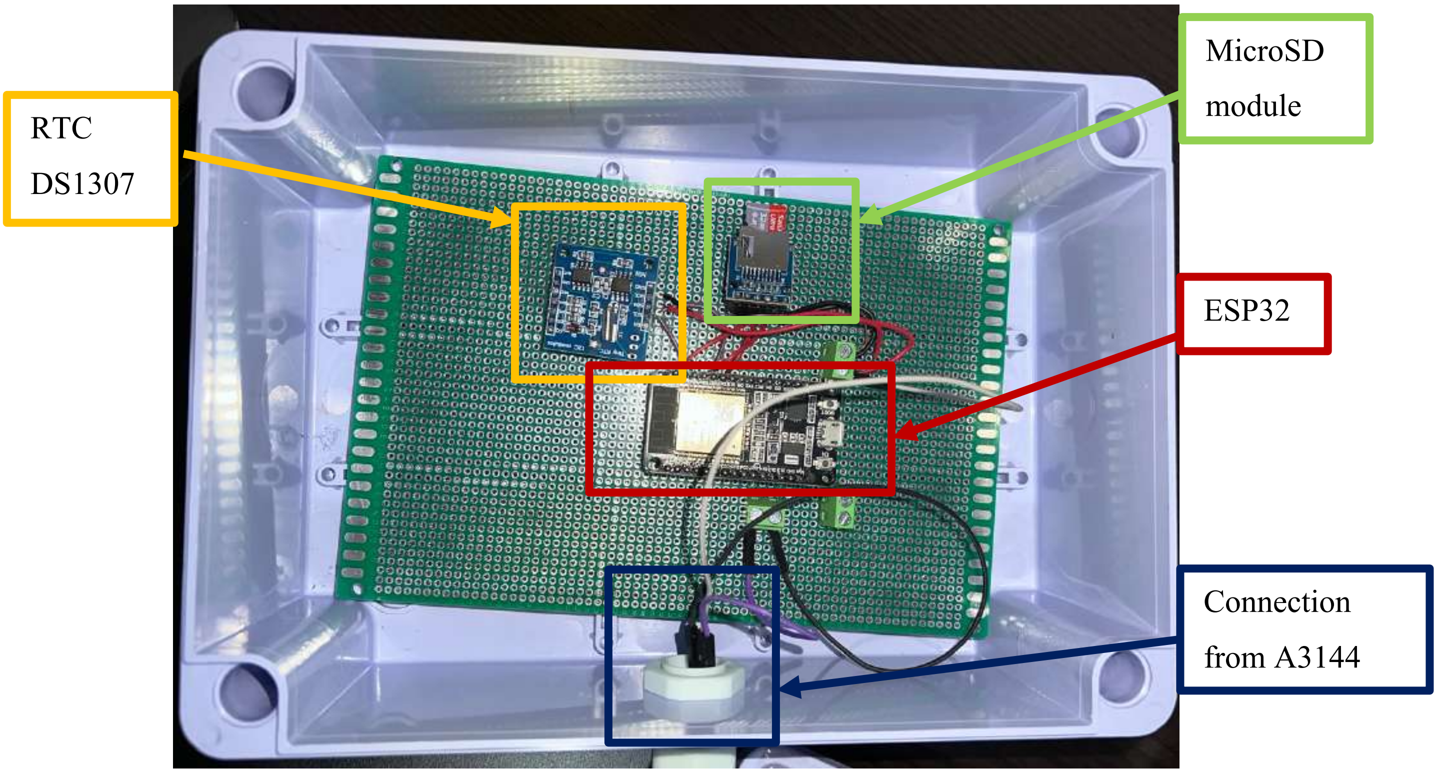

Figure 3.25: Finalised Circuit assembled in the Electrical Junction Box

3.6 Project Costing

Table 3.10: Overall Project Costs

Components/ Materials	Quantity	Price per Unit	Total Price
Electrical Junction Box	1	RM12.70	RM12.70
Protoboard/ Donut Board (12 x 18 cm)	1	RM17.10	RM17.10
PCB Screw Connector	3	RM0.80	RM2.40
Rain Gauge Frame	1	~RM80	RM80
ESP32	1	RM20.80	RM20.80
RTC DS1307	1	RM2.90	RM2.90
MicroSD Card	1	RM16.00	RM16.00
MicroSD Card Module	1	RM1.10	RM1.10
Cable Gland	1	RM0.65	RM0.65
Total:			RM153.65

The overall total cost for this project considering only the components and materials that is used directly in this project is only RM153.65. The average price of a tipping bucket rain gauge that is as small as this size that are available in the market are at least RM4,000, making the tipping bucket rain gauge that is developed in this research project to be affordable compared to it.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter encompasses the results that are obtained from the tipping bucket rain gauge that has been developed over the course of two trimesters. The outcome from the results will be briefly discussed and analyse with comparison with an industrial rain gauge from a nearby weather station. However, since the project period is limited to two trimesters, there is a time limitation for the collection of data due to most of the time being used up on developing the hardware, drafting the circuits and testing the circuits functionality and capability of the measuring unit to collect data and also the capability of the prototype system to send the data to cloud for easy access and visualization of the data. Hence, the results obtained from the rain gauge spans over four days, starting from 4th of April 2024.

4.2 Deployment Setup of the Tipping Bucket Rain Gauge

The location of deployment of the tipping bucket rain gauge prototype is at the rooftop of UTAR Sungai Long Campus KB Block.

Figure 4.1: Deployment of Tipping Bucket Rain Gauge

Figure 4.2: Another view of the Deployement of the Rain Gauge

The developed prototype is placed at a strategic location where the objects around it do not affect the measuring of the rain rate. The prototype is made sure to be placed at an open area and an area that is big enough that the object surrounding it will not affect the collection of the data. The reason why the location for the deployment is important because if the prototype is placed at a wrong location, the data obtained might be inaccurate. For example, if the prototype is placed beside a wall, the rain water might splash and reflect from the wall and dripped into the opening area of the rain gauge, making the rain data to be affected and providing inaccurate results.

4.3 Output File Format

The rain gauge prototype is configured to run for 24 hours daily and sends output data to the cloud every minute. The output data is sent into a Google Sheets spreadsheet with the file name of “Rain_Rate”. From Google Sheets, the data can be downloaded directly in different forms including Microsoft Excel (.xlsx) format, Comma-Seperated Value (CSV) format, PDF (.pdf) format and many more. The spreadsheet starts with a header of date, time, rain rate, and weather (accumulated). The date and time is stamped by the RTC module used in the circuit and is sent to the spreadsheet along with the rain rate data.

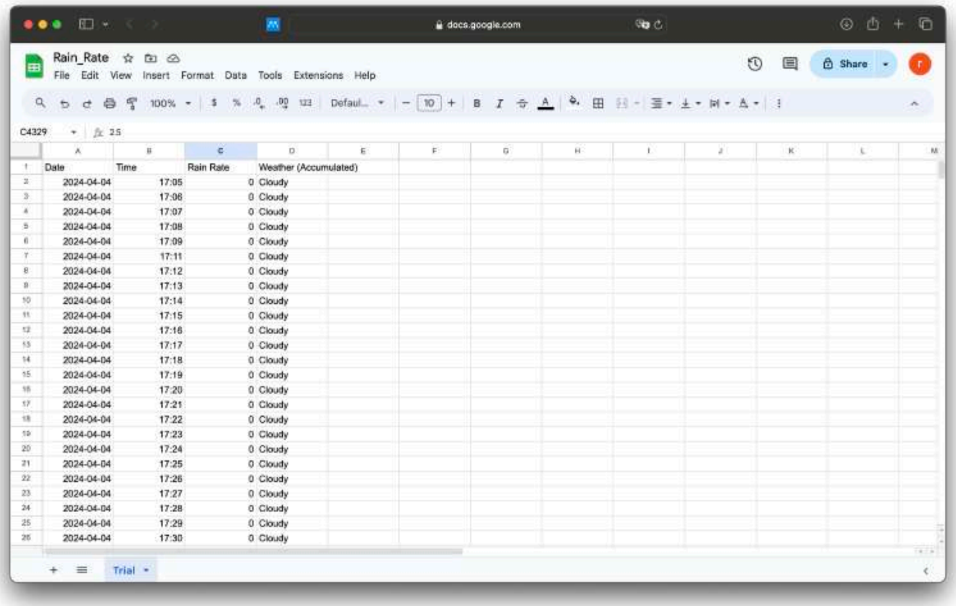

Figure 4.3: Partial Screenshot of the Google Sheets spreadsheet

Next, alongside the data being sent to Google Sheets, the data is also saved to the local MicroSD card as a backup. The same parameters are added into the MicroSD card. However, the data that is sent and saved to the MicroSD card is in the form of text file, separated by commas, making it easier to obtain the file and convert it into a csv file for better processing and visualization of the data. The conversion of the file from .txt to .csv can be easily done by the use of Python codes and also through dedicated online websites that converts text file into csv file.

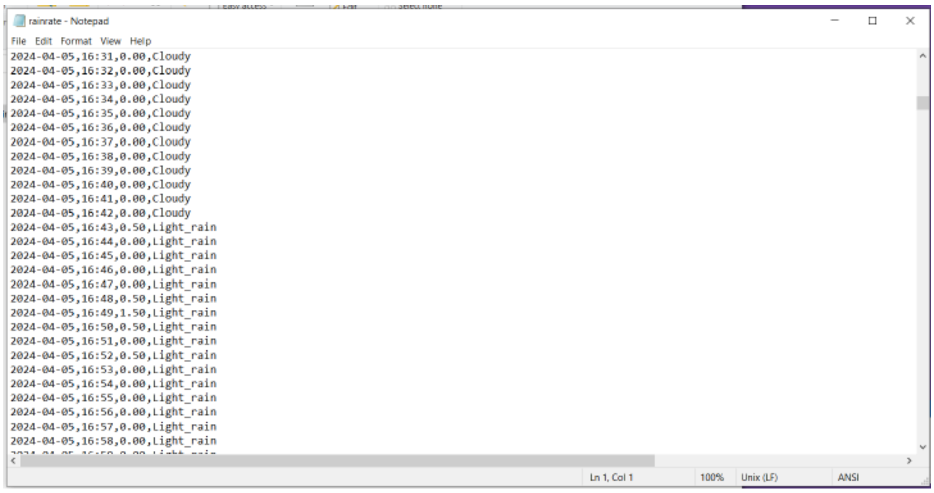

Figure 4.4: Screenshot of Text File

4.4 Collected Data of Rain Rate

Over the time when the prototype is deployed on the rooftop area, it rains on 5th of April 2024 and 7th of April 2024. Hence, the discussion and analysis on the collected data will be based on the data collected from these two days, since the other time that is not raining shows the data of 0 consistently. The data collected will be compared with the rain rate data from a nearby station, which is the Station Sg. Balak in Kg. Baru Balakong which is located around 6-7 km from the deployment location of the rain gauge prototype.

4.4.1 Analysis of Rain Rate (5th of April 2024)

The rain gauge prototype first picked up the first tipping at 14:49 or 2:49 PM, and the last tipping from the prototype is at 17:38 or 5:38 PM. Hence, the analysis of the data will be in the timeframe of 14:40 or 2:40 PM to 18:00 or 6:00 PM.

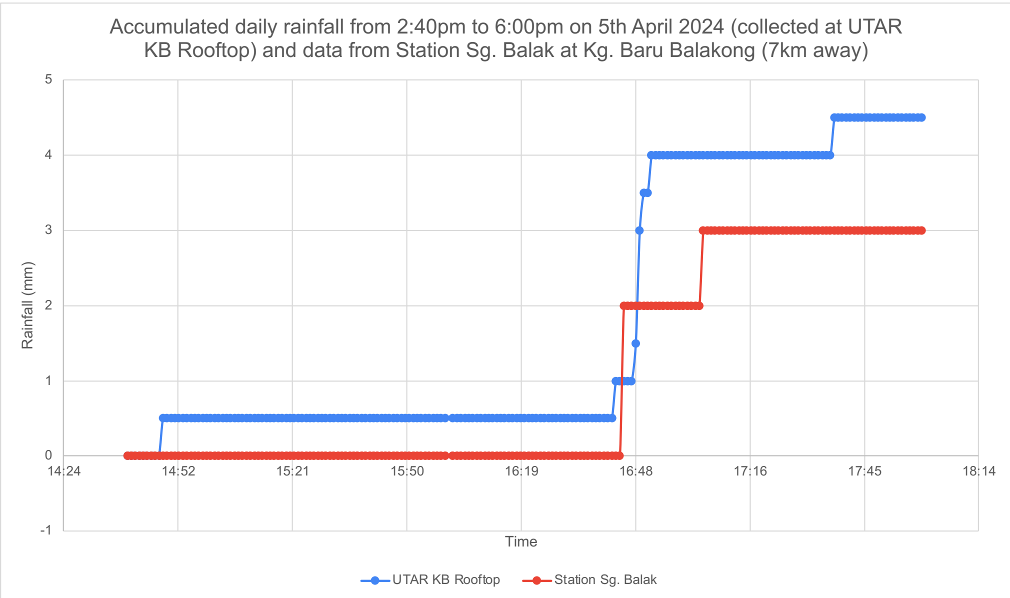

Figure 4.5: Comparison of Rainfall between Prototype and Data from Station Sg. Balak

From the above figure, it can be seen that there are differences between both of the data. The accumulated daily rainfall for the station in Sg. Balak on that day is 3 mm while the rain gauge recorded 4.5 mm on that day. At 2:49 PM, the prototype recorded a single tip of rain, contributing to 0.5 mm of rain at that time. Then at 4:43 PM and 4:48 PM, the prototype also recorded a single tip of rain each, contributing to a total of 1 mm of rain, at the next minute, on 4:49 PM, the prototype recorded 1.5 mm of rain, which is the result of 3 tips and 1 tip at 4:50 PM. At 4:52 PM, it recorded another tip of rain and finally at 5:38 PM, the prototype recorded the final tip of the day, contributing to a total of 4.5 mm being recorded on that day.

For the data from Sg. Balak station, the first recorded rain amount is at 4:45 PM with an amount of 2 mm being recorded. Then, the next recorded time is at 5:15 PM, with an amount of 1 mm, making the total accumulated rainfall on that day to be 3 mm.

For the comparison of rain rate per hour, taking the timeframe of 4:00 PM to 5:00 PM, the rain rate is 3.5 mm/hr for the prototype while for the rain rate from Sg. Balak station is 2.0 mm/hr. Then, for the timeframe of 5:00 PM to 6:00 PM, the rain rate from the prototype is 0.5 mm/hr and the rain rate from Sg. Balak station is 1.0 mm/hr.

Using all of the above values, we can calculate the percentage difference between both data:

1. Difference in Accumulated Rainfall per day:

$$Prototype\ recorded\ rainfall = 4.5\ mm/day$$

$$Data\ from\ Sg.Balak\ Station = 3.0\ mm/day$$

$$Percentage\ difference\ between\ data = \frac{|4.5 - 3.0|}{\frac{1}{2}(4.5 + 3.0)} \times 100\% = 40\%$$

2. Difference in rain rate:

   At 3:00 PM to 4:00 PM:

$$Percentage\ difference = \frac{|0.5 - 0|}{\frac{1}{2}(0.5 + 0)} \times 100\% = 200\%$$

   At 4:00 PM to 5:00 PM:

$$Percentage\ difference = \frac{|3.5 - 2.0|}{\frac{1}{2}(3.5 + 2.0)} \times 100\% = 54.55\%$$

   At 5:00 PM to 6:00 PM:

$$Percentage\ difference = \frac{|0.5 - 1.0|}{\frac{1}{2}(0.5 + 1.0)} \times 100\% = 66.67\%$$

$$Average\ percentage\ difference = \frac{200 + 54.55 + 66.67}{3} = 107\%$$

From the calculation above, the obtained percentage difference in accumulated rainfall per day is 40% whereas the average percentage difference in rain rate is 107%, this shows the percentage difference between the two data to be very big. However, at the time of 2:49 PM, at the actual location, no rain were seen or observed physically when the rain gauge was deployed on the rooftop. The tipping mechanism might be wrongfully triggered due to the big wind or other external factors. Hence, if we remove the abnormal data from the analysis, the comparison will be changed to as below, where the timeframe is from 4:40 PM to 6:00 PM.

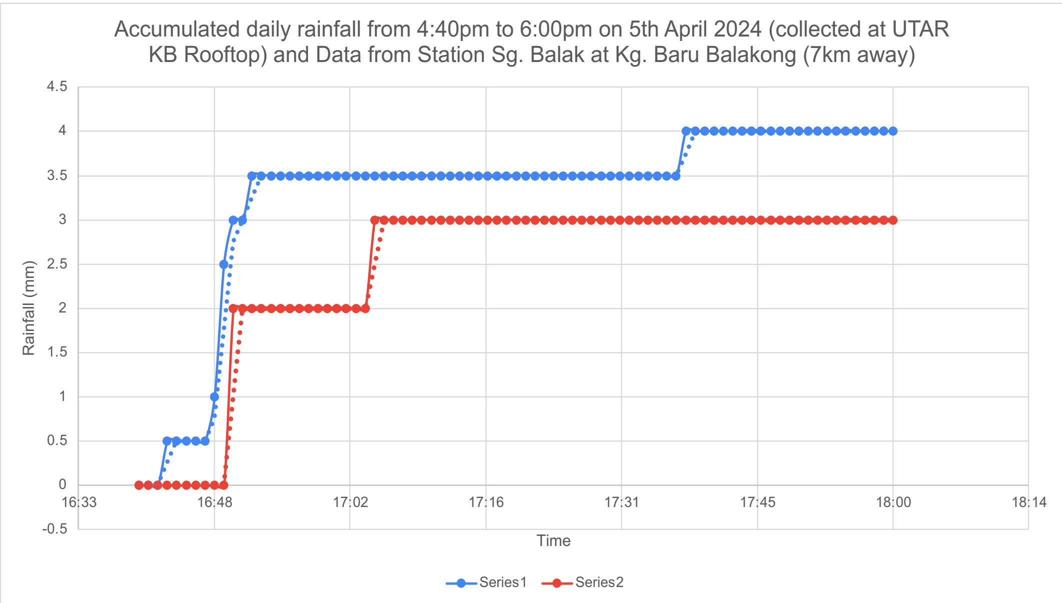

Figure 4.6: Comparison of Rainfall between Prototype and Data from Station Sg. Balak (without anomalies)

From here we can see that the difference of the accumulated daily rainfall between both data has been reduced while the difference in rain rate stays the same for timeframe of 4:00 PM to 5:00 PM and 5:00 PM to 6:00 PM. The total daily rainfall recorded from the prototype is now 4 mm while the data from the station stays the same at 3 mm. From the obtained results, the percentage difference can be calculated as below:

1. Difference in accumulated rainfall per day:

$$Percentage\ difference = \frac{|4-3|}{\frac{1}{2}(4+3)} \times 100\% = 28.57\%$$

2. Difference in rain rate:

At 4:00 PM to 5:00 PM:

$$Percentage\ difference = \frac{|3.5-2.0|}{\frac{1}{2}(3.5+2.0)} \times 100\% = 54.55\%$$

At 5:00 PM to 6:00 PM:

$$Percentage\ difference = \frac{|0.5-1.0|}{\frac{1}{2}(0.5+1.0)} \times 100\% = 66.67\%$$

$$Average\ percentage\ difference = \frac{54.55+66.67}{2} = 60.61\%$$

It is known that the difference between the accumulated rainfall per day data is only 1 mm but the difference between the rain rate data at 4:00 PM to 5:00 PM is 1.5 mm/hr while at 5:00 PM to 6:00 PM, the difference is 0.5 mm/hr.

However, another factor to consider is the distance between the two data collection point, one is at UTAR KB Rooftop while the other one is at Sg. Balak Station at Kg. Baru Balakong. The distance between two points is around 7 km, making the data between both of the location to be different as the cover area of the cloud may not reached to the same location, making the intensity of the rainfall to be different, hence the difference in data.

From the prototype data, the rain rate at 4:00 PM to 5:00 PM is 3.5 mm/hr while the rain rate at 5:00 PM to 6:00 PM is 0.5 mm/hr. Using these two data, we can obtain the rain fade at these instances using the equation from the ITU-R P.838 model illustrated in the equation (2.4.

Table 4.1: Table of Specific Attenuation at Different Frequency Levels

Frequency (GHz)	4:00 PM – 5:00 PM (R = 3.5 mm/hr)		5:00 PM – 6:00 PM (R = 0.5 mm/hr)	
	γ_R (horizontal)	γ_R (vertical)	γ_R (horizontal)	γ_R (vertical)
10	0.0588	0.0518	0.0051	0.0049
20	0.3444	0.3300	0.0441	0.0486
30	0.7885	0.7190	0.1245	0.1217
40	1.3133	1.2274	0.2429	0.2384
50	1.8171	1.7349	0.3769	0.3751
60	2.2456	2.1664	0.5062	0.5079
70	2.5887	2.5316	0.6200	0.6218
80	2.8539	2.8118	0.7147	0.7172
90	3.0567	3.0279	0.7914	0.7944
100	3.2106	3.1926	0.8524	0.8559

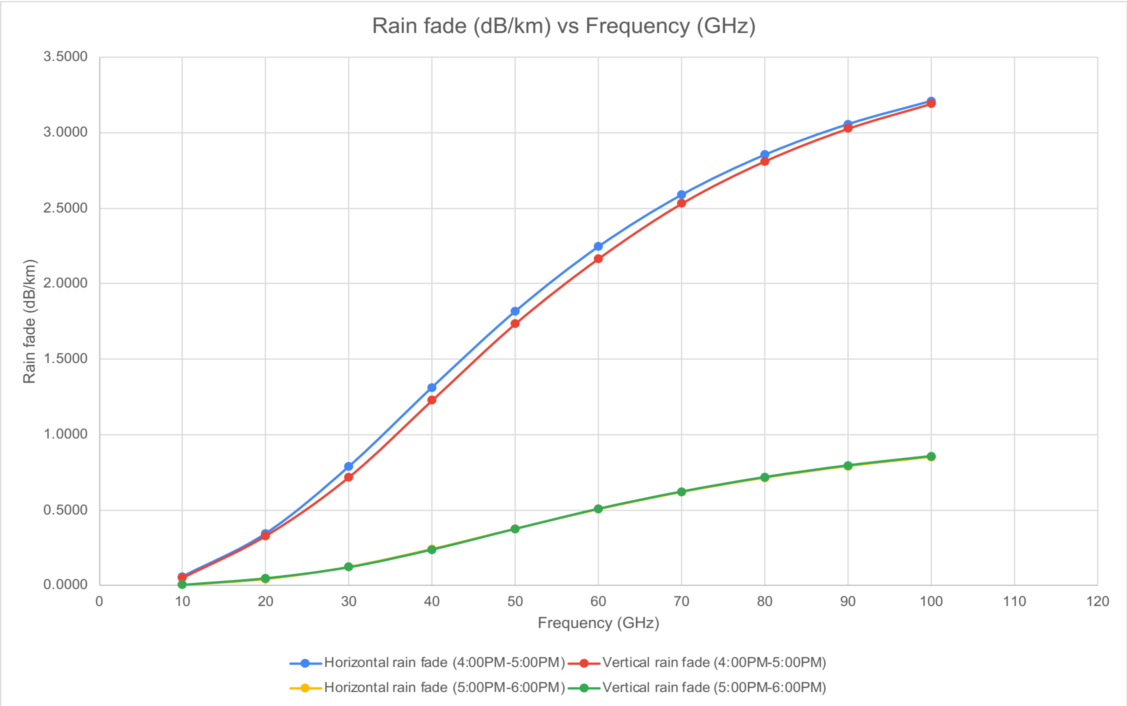

Figure 4.7: Graph of Rain Fade (dB/km) vs Frequency (GHz)

From the above graph, it is observed that there are no big difference between the horizontal polarization rain fade and the vertical polarization rain fade. The more prominent difference can be seen is the difference of rain fade between higher rain rate and lower rain rate. At 4:00 PM – 5:00 PM, when the rain rate is 3.5 mm/hr, the rain fade can be seen to be much higher than the rain fade at 0.5 mm/hr. From here, we can conclude that when the rain rate increases, the specific rain attenuation or rain fade will increase too. It is also observed that the rain fade is having a logistic growth rate and when the rain fade reaches a certain threshold or value, the effect of the increase in frequency does not affect much on the rain fade anymore.

4.4.2 Analysis of Rain Rate (7th of April 2024)

The timeframe for the analysis and visualisation of the data is from 5:00 PM to 6:00 PM as rainfall is only recorded within this time period. The graph of rainfall against time is as below:

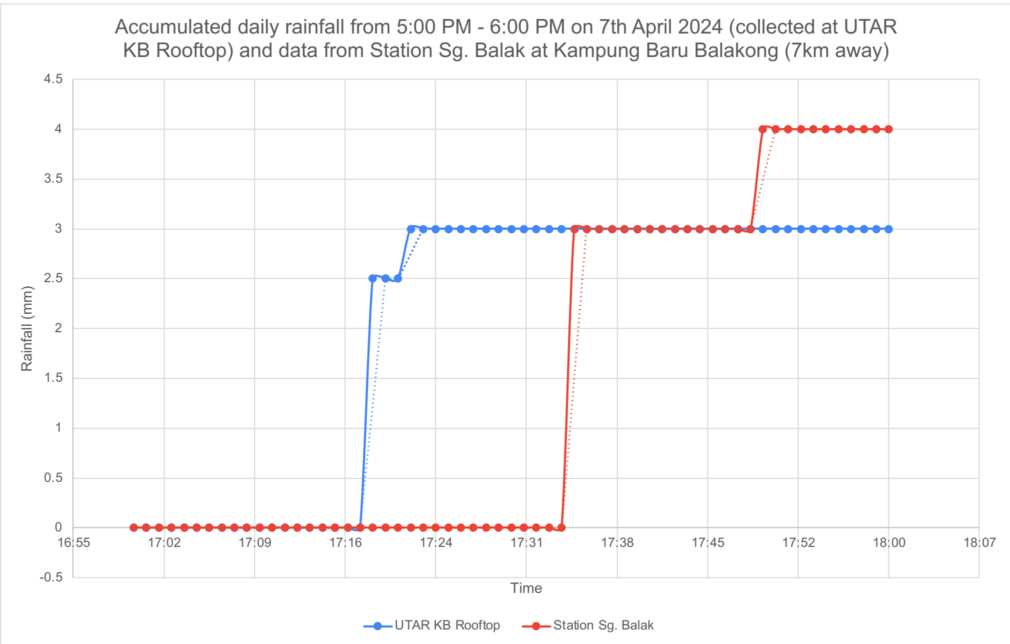

Figure 4.8: Comparison of Rainfall between Prototype and Data from Station Sg. Balak

From the graph above, it is seen that the rain gauge prototype recorded the first rain data from the rainfall at 5:19 PM which the prototype tips 5 tips, representing 2.5 mm of rainfall. Then, three minutes later, at 5:22 PM, the prototype receive a single tip, which is 0.5 mm and it brings the total rainfall accumulated that day to 3 mm. For the data from station Sg. Balak, the rain is first detected at 5:35 PM, which gives the data of 3 mm. Then, the data shows another 1 mm of rain at 5:50 PM, making the total accumulated rainfall for the day to be 4 mm. The difference between the data may be due to the reason that was previously discussed, which is the location of the data collection. From the graph, the rain rate for 5:00 PM to 6:00 PM can also be determined to be 3 mm/hr for the prototype and 4 mm/hr for the station Sg. Balak. The calculation of the percentage difference is as below:

Difference in accumulated rainfall per day and difference in rain rate :

$$Percentage\ difference = \frac{|4-3|}{\frac{1}{2}(4+3)} \times 100\% = 28.57\%$$

From the obtained results, the rain fade graph can be calculated using the rain rate of 3 mm/hr at 5:00 PM to 6:00 PM.

Table 4.2: Table of Specific Attenuation at Different Frequency Levels

Frequency (GHz)	5:00 PM – 6:00 PM (R = 3 mm/hr)	
	γ_R (horizontal)	γ_R (vertical)
10	0.0484	0.0429
20	0.2926	0.2835
30	0.6812	0.6246
40	1.1490	1.0780
50	1.6042	1.5367
60	1.9957	1.9312
70	2.3116	2.2651
80	2.5574	2.5234
90	2.7464	2.7234
100	2.8904	2.8765

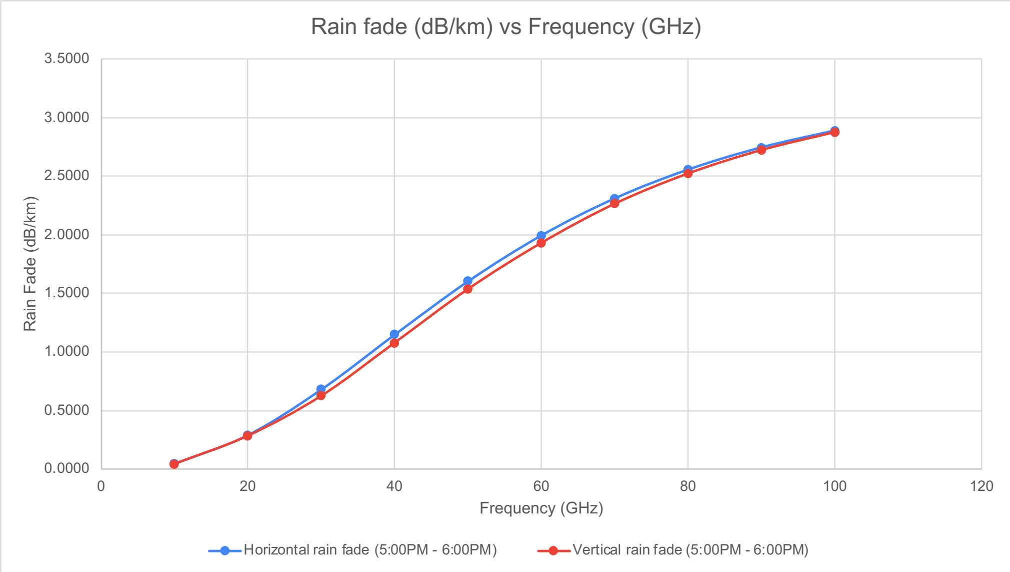

Figure 4.9: Graph of Rain Fade (dB/km) vs Frequency (GHz)

4.5 Summary of Results and Discussions

The collection of rain rate data started from 4th of April 2024 when thorough verification of the rain gauge prototype are done. Due to the project period is limited to two trimesters only, the time were mostly used up on the realisation of the hardware and testing the circuits and the capability of the measuring unit to collect accurate data and also testing the coding for the flow of the whole prototype system. Hence, the data collection only starts on 4th of April 2024.

As mentioned, the placement location of the prototype to collect data was considered thoroughly and include external factors that may cause the data collected to be inaccurate. The prototype was deployed on an open area and the prototype is placed at the highest possible point and far from walls and objects that may cause the rain water to splash into the tipping bucket rain gauge, causing inaccuracy of the data.

Furthermore, the results obtained on both days of the analysis shows disrespancies with the data from the nearest rain collection center, at Sungai Balak Station in Kampung Baru Balakong. However, the distance between the location of deployment of the rain gauge and the station should be taken into account since the distance between them is 7 km away. Hence, the data taken from the station can still be used as a reference for the prototype that was developed.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion of the Project Outcome

In conclusion, the main goal of developing an affordable rain gauge for rain rate and fade measurements has been achieved. A microcontroller-based rain gauge that is low-cost, affordable and capable of obtaining reliable rain rate data has been designed and constructed. The rain gauge uses the tipping bucket mechanisms and is also able to operate autonomously with the minimum amount of human interaction to the prototype. The features of the rain gauge that was developed includes:

1. Tipping bucket mechanism that uses a magnet and a Hall Effect Sensor that senses magnetic flux and providing output to detect tips and converting it to the rainfall data.
2. ESP32 that acts as a brain of the circuit and controls the entire circuit including signal processing, receiving and sending of data to different parts of the circuit and also storage of data.
3. ESP32 ensures the system is connected to the WiFi before uploading to the cloud (Google Sheets) for data saving.
4. MicroSD card connected externally to save the data if the ESP32 is not connected to WiFi to ensure no data loss.
5. Timestamp that is logged by the external real-time clock module for accurate datalogging process.
6. The interval for uploading the data to cloud is set to every 1 minute, making the data recorded to be rain rate per minute.
7. The protoboard with the components are sealed within an electrical junction box that provides waterproofing to the entire circuit system to prevent tripping of electricity.

5.2 Limitations and Recommendations for Future Improvement

Over the course of designing and implementing the hardware prototype, many factors that limits and slows down the process to develop the final working prototype has been met. As discussed, most of the time were used up in designing and setting up the hardware, testing the hardware and also developing the coding for the prototype, the data collection can only be done for a short period of time, making the verification of the data cannot be done for a long period of time. Therefore, it is recommended that the prototype is continued to be run and more verification of data is needed to make sure that the prototype is capable of reading data and operate for a long time.

Next, the previous recommendations leads to the second limitation of this project, which is regarding the capability of the prototype to run for a long time. Although the prototype's circuit runs perfectly without causing any issue and is able to upload the data for every minute without fail, the problem that causes the prototype to be stopped working is the 3D printed rain gauge itself, which is the most important part of the system. The 3D printed material used in this project is PLA+ which is known to be cheap and able to withstand heat, however, upon further investigation and research on the material, it is found that the PLA+ material used is not as resistant to heat as it is known. After deploying the prototype for a prolonged period of time, the body of the rain gauge has deformed due to being exposed to high heat for too long, causing the tipping bucket to not be able to operate due to the tipping bucket being stucked. The recommendation for future works to prevent this issue from happening again is to use better heat resistant material as the prototype will be placed in the Sun for a long period of time.

Thirdly, the next limitation is due to the size and weight of the prototype itself. Due to the nature of the material used for the prototype, the size and weight of the prototype is very light, hence, causing the prototype to be prone to being blown away to the wind. When the prototype is deployed, the cover, which is the opening area of the rain gauge flown off and was blown away by the wind due to the prototype being too light. Hence, for future work, it is recommended to glue the cover of the opening area to the body of the rain gauge, the material can also be replaced with heavier materials too.

REFERENCES

Acharya, R., 2017. Tropospheric impairments: Measurements and mitigation. *Satellite Signal Propagation, Impairments and Mitigation*, pp.195–245.

Alozie, E. et al., 2022. A Review on Rain Signal Attenuation Modeling, Analysis and Validation Techniques: Advances, Challenges and Future Direction. *Sustainability 2022, Vol. 14, Page 11744*, 14(18), p.11744. Available at: https://www.mdpi.com/2071-1050/14/18/11744/htm [Accessed: 3 July 2023].

American Meteorological Society, 2022, *Rainfall rate - Glossary of Meteorology* [Online]. Available at: https://glossary.ametsoc.org/wiki/Rainfall_rate [Accessed: 3 July 2023].

Anon, *Difference between Arduino vs ESP 32* [Online]. Available at: https://www.simplyiotsensors.com/2023/05/Arduino-vs-ESP-32.html# [Accessed: 9 September 2023a].

Anon, *ESP32 Pinout Reference - Last Minute Engineers* [Online]. Available at: https://lastminuteengineers.com/esp32-pinout-reference/ [Accessed: 18 April 2024b].

Anon, *ESP32 SPI Communication: Pins, Multiple SPI, Peripherals (Arduino) | Random Nerd Tutorials* [Online]. Available at: https://randomnerdtutorials.com/esp32-spi-communication-arduino/ [Accessed: 18 April 2024c].

Anon, *eXe* [Online]. Available at: https://courses.comet.ucar.edu/pluginfile.php/3721/mod_imscp/content/1/reflectivityrainfall_relationship.html [Accessed: 13 July 2023d].

Anon, 2022, *Hall Effect Sensor A3144 Magnetic Switch Basics Working* [Online]. Available at: https://how2electronics.com/hall-effect-sensor-a3144-magnetic-switch/# [Accessed: 19 April 2024].

Anon, *How Does A Tipping Bucket Rain Gauge Work? -Renke* [Online]. Available at: https://www.renkeer.com/tipping-bucket-rain-gauge-principle/# [Accessed: 12 September 2023e].

Anon, *Rainfall Data – The Official Web of Public Infobanjir* [Online]. Available at: https://publicinfobanjir.water.gov.my/hujan/data-hujan/?lang=en [Accessed: 16 April 2024f].

Arathy, J., 2021, *Complete guide on Real-Time Clock RTC DS1307 Module* [Online]. Available at: https://www.electrorules.com/complete-guide-on-real-time-clock-rtc-ds1307-module/ [Accessed: 9 September 2023].

Barani, J., 2020, *Rain rate intensity classification — BARANI* [Online]. Available at: https://www.baranidesign.com/faq-articles/2020/1/19/rain-rate-intensity-classification [Accessed: 3 July 2023].

Britannica, *Rain attenuation | Britannica* [Online]. Available at: https://www.britannica.com/technology/rain-attenuation [Accessed: 4 July 2023].

CementConcrete, 2020, *Rain Gauge: Uses, Types, diagram, rainfall measurement, Data Adjustment & site Selection - Cement Concrete* [Online]. Available at: https://cementconcrete.org/water-resources/hydrology/rain-gauge/2637/ [Accessed: 14 July 2023].

Columbia Weather Systems, *Rain/Precipitation Measurement Technologies | Columbia Weather Systems* [Online]. Available at: https://columbiaweather.com/resources/rain-measurement/ [Accessed: 14 July 2023].

Fashuyi, M.O. and Afullo, T.J., 2007. Rain attenuation prediction and modeling for line-of-sight links on terrestrial paths in South Africa. *Radio Science*, 42(5), p.n/a-n/a.

Fournier, J., 1999, *REFLECTIVITY-RAINFALL RATE RELATIONSHIPS IN OPERATIONAL METEOROLOGY* [Online]. Available at: https://www.weather.gov/tae/research-zrpaper# [Accessed: 13 July 2023].

Freeman, L., 2021, *How to collect rainwater in your own backyard* [Online]. Available at: https://spectrumnews1.com/ca/la-west/weather/2021/02/18/measuring-rain-in-your-own-backyard---we-ve-got-you-covered [Accessed: 10 August 2023].

Gannon, P., *What is Rain Fade? - BusinessCom Networks* [Online]. Available at: https://www.bcsatellite.net/blog/what-is-rain-fade/ [Accessed: 10 August 2023].

Hyde, G. and Bargellini, P.L., 2002. Satellite and Space Communications. *Reference Data for Engineers*, pp.27–1.

Hydreon, *Home | Rain Sensors* [Online]. Available at: https://rainsensors.com/ [Accessed: 26 July 2023].

Kent, *Cloud Storage vs Local Storage: Which is the Better Option? - Document Management System Folderit* [Online]. Available at: https://www.folderit.com/blog/cloud-storage-vs-local-storage/ [Accessed: 9 September 2023].

Maximum Weather Instruments, 2020, *What are the Different Types of Rain Gauges? | Maximum Learning Center* [Online]. Available at: https://www.maximum-inc.com/what-are-the-different-type-of-rain-gauges/ [Accessed: 14 July 2023].

Meaney, D., *What is a Real Time Clock (RTC)? - ECS Inc.* [Online]. Available at: https://ecsxtal.com/what-is-a-real-time-clock-rtc/ [Accessed: 19 April 2024].

NASA, *CoCoRaHS Rain Gauge Before a Storm | NASA Global Precipitation Measurement Mission* [Online]. Available at: https://gpm.nasa.gov/image-gallery/cocorahs-rain-gauge-storm [Accessed: 10 August 2023].

Olsen, R.L., Rogers, D. V. and Hodge, D.B., 1978. The aRb Relation in the Calculation of Rain Attenuation. *IEEE Transactions on Antennas and Propagation*, 26(2), pp.318–329.

Pachori, A., 2022, *Google Sheets Max Rows? 10 Million Cells Is The Limit* [Online]. Available at: https://www.gigasheet.com/post/google-sheets-limits-10-million-cells# [Accessed: 17 April 2024].

Radiocommunication Bureau, I., 1992. RECOMMENDATION ITU-R P.838-3 - Specific attenuation model for rain for use in prediction methods.

Samad, M.A., Diba, F.D. and Choi, D.Y., 2021. A Survey of Rain Fade Models for Earth–Space Telecommunication Links—Taxonomy, Methods, and Comparative Study. *Remote Sensing 2021, Vol. 13, Page 1965*, 13(10), p.1965. Available at: https://www.mdpi.com/2072-4292/13/10/1965/htm [Accessed: 4 July 2023].

Scientific Sales, *Orion LX Weather Station | Industrial Weather Stations* [Online]. Available at: https://www.scientificsales.com/Orion-LX-Weather-Station-with-Display-Console-p/9610-c-1.htm [Accessed: 25 July 2023].

Segovia-Cardozo, D.A., Bernal-Basurco, C. and Rodríguez-Sinobas, L., 2023. Tipping Bucket Rain Gauges in Hydrological Research: Summary on Measurement Uncertainties, Calibration, and Error Reduction Strategies. *Sensors 2023, Vol. 23, Page 5385*, 23(12), p.5385. Available at:

https://www.mdpi.com/1424-8220/23/12/5385/htm [Accessed: 10 August 2023].

Teja, R., 2021, *What are the differences between Raspberry Pi and Arduino?* [Online]. Available at: https://www.electronicshub.org/raspberry-pi-vs-arduino/#Differences_between_Raspberry_Pi_and_Arduino [Accessed: 9 April 2023].

Tessie, 2021, *A3144 Hall Effect Sensor: Datasheet, Circuit and Pinout* [Online]. Available at: https://www.utmel.com/components/a3144-hall-effect-sensor-datasheet-circuit-and-pinout?id=828 [Accessed: 9 September 2023].

The History of Rain Measurement, THE HISTORY OF RAIN MEASUREMENT. Available at: www.islandnet.com/~see/weather/history [Accessed: 14 July 2023].

Weather Shack, *Tipping Bucket Rain Gauge* [Online]. Available at: https://www.weathershack.com/static/ed-tipping-bucket-rain-gauge.html [Accessed: 15 July 2023].

APPENDICES

Appendix A: Arduino Coding for ESP32

```
//Importing Libraries
#include <Wire.h>
#include "RTClib.h"
#include "WiFi.h"
#include <WiFiClientSecure.h>
#include <HTTPClient.h>
#include <FS.h>
#include <SD.h>

RTC_DS1307 rtc; //Real time clock module used is DS1307 module
DateTime now;

//Initialising constants and variables
const int pin_interrupt = 14; //GPIO14 on ESP32 set as interrupt pin
long int no_of_tips = 0; //Recording number of tips
long int temp_no_of_tips = 0; //Temporary variables

float rain_rate = 0.00;

float rain_rate_per_minute = 0.00;
float temp_rain_rate_per_minute = 0.00;

float rain_rate_per_hour = 0.00;
float temp_rain_rate_per_hour = 0.00;

float rain_rate_per_day = 0.00;
float temp_rain_rate_per_day = 0.00;

float today_rain_rate = 0.00;
float mm_per_tip = 0.5; //One tip equals 0.5 mm
```

```
String weather = "Cloudy";

volatile boolean flag = false;

//Initialising date and time variables
String date, month, year, hour, minute, second, actual_year;
int temp_year;

//WiFi credentials
char ssid[] = "Netcore";
char password[] = "netcore12345678";

//Google Apps Script ID
String GAS_ID =
"AKfycbz76vSVu28W70T2zrZCiV10FpoKJPtv8B5Q2DcWuWNTIJCU7E
9canoomVbB3lh_1ijmJA";

//const char* host = "script.google.com";
//const int httpsPort = 443;

WiFiClientSecure client;

//Counting tips
void ICACHE_RAM_ATTR count_rain_rate()
{
	flag = true;
}

//Setting up the prototype
void setup()
{
	Serial.begin(9600); //data exchange rate 9600 bits per second
```

```
delay(500);

connectToWiFi();

        pinMode(pin_interrupt, INPUT); //Setting GPIO14 as input pin

        attachInterrupt(digitalPinToInterrupt(pin_interrupt),
count_rain_rate, FALLING); //Count tips as pin from logic HIGH to LOW

        //Initialising RTC module - checking RTC connection
        if (!rtc.begin())
        {
                Serial.println("Couldn't find RTC");
                while (1);
        }

        //rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //Obtaining
the date and time from PC and adjust to RTC module

//Initializing SD Card
if (!SD.begin())
{
  Serial.println("SD Card Initialization failed!");
}

uint8_t cardType = SD.cardType();

if (cardType == CARD_NONE)
{
  Serial.println("No SD Card!");
}

//Print card type
```

```
  Serial.print("SD Card Type: ");
  if (cardType == CARD_MMC)
  {
    Serial.println("MMC");
  }
  else if (cardType == CARD_SD)
  {
    Serial.println("SDSC");
  }
  else if (cardType == CARD_SDHC)
  {
    Serial.println("SDHC");
  }
  else
  {
    Serial.println("UNKNOWN");
  }

        readRTC();
        printSerial();
  client.setInsecure();
}

//Looping and reading rain_rate
void loop()
{
        if (flag == true)
        {
                rain_rate += mm_per_tip; //rain_rate increases after every
tip
                no_of_tips++;
                delay(500);
                flag = false; //reset flag for another count
```

```
}

readRTC();
today_rain_rate = no_of_tips * mm_per_tip;
temp_rain_rate_per_minute = rain_rate;

//https://publicinfobanjir.water.gov.my/hujan/data-hujan/?lang=en
if (today_rain_rate <= 0.00 && today_rain_rate < 1.00)
{
        weather = "Cloudy";
}

if (today_rain_rate >= 1.00 && today_rain_rate <= 10.00)
{
        weather = "Light_rain";
}

if (today_rain_rate > 10.00 && today_rain_rate <= 30.00)
{
        weather = "Moderate_rain";
}

if (today_rain_rate > 30.00 && today_rain_rate <= 60.00)
{
        weather = "Heavy_rain";
}

if (today_rain_rate > 60.00)
{
        weather = "Very_Heavy_rain";
}

//rain rate per minute, hour, and day calculated at second = 0
```

```
            if (second.equals("0"))
            {
                        rain_rate_per_minute = temp_rain_rate_per_minute;
                        temp_rain_rate_per_hour += rain_rate_per_minute; // rain
rate per hour is calculated from total rain rate per minute, but stored in
temporary variable first

                        if (minute.equals("0")) //Only count rain rate per hour at
minute = 0
                        {
                                    rain_rate_per_hour = temp_rain_rate_per_hour;
                                    temp_rain_rate_per_day += rain_rate_per_hour;
// rain rate per day is calculated from total rain rate per hour, but stored in
temporary variable first

                                    temp_rain_rate_per_hour = 0.00; //Reset
temp_rain_rate_per_hour
                        }

                        if (minute.equals("0") && hour.equals("0")) //12 midnight
                        {
                                    rain_rate_per_day = temp_rain_rate_per_day;
                                    //Reset variables
                                    temp_rain_rate_per_day = 0.00;
                                    today_rain_rate = 0.00;
                                    no_of_tips = 0;
                        }
                        //Reset variables
                        temp_rain_rate_per_minute = 0.00;
                        rain_rate = 0.00;
                        delay(1000);
            }
```

```
        if ((no_of_tips != temp_no_of_tips) || (second.equals("0"))) //Print
serial every 1 minute or when the number of tips changes
        {
                printSerial();
        }
        temp_no_of_tips = no_of_tips;
}

//Function to format time display
String time_conversion(String number)
{
        if (number.length() == 1)
        {
                number = "0" + number; //Adding a 0 in front of single
digit number
        }
        else
        {
                number = number;
        }
        return number;
}

//Function to read RTC module
void readRTC()
{
        DateTime now = rtc.now(); //Obtain date time from DS1307
        date = String(now.day(), DEC);
        month = String(now.month(), DEC);
        temp_year = now.year() - 2000;
  actual_year = String(now.year(), DEC);
        year = String(temp_year);
        hour = String(now.hour(), DEC);
```

```
            minute = String(now.minute(), DEC);
            second = String(now.second(), DEC);
}

//Function to print information on Serial Monitor
void printSerial()
{
  String d = time_conversion(actual_year) + "-" + time_conversion(month) +
"-" + time_conversion(date) ; // date
  Serial.println(d);
  String t = time_conversion(hour) + ":" + time_conversion(minute) ; // time
            Serial.println(t);
            Serial.print("Weather =");
            Serial.println(weather);
            Serial.print("No of tips =");
            Serial.print(no_of_tips);
            Serial.println(" times ");
            Serial.print("Today rain rate =");
            Serial.print(today_rain_rate, 2);
            Serial.println(" mm ");
            Serial.print("Rain rate per minute =");
            Serial.print(rain_rate_per_minute, 2);
            Serial.println(" mm ");
            Serial.print("Rain rate per hour =");
            Serial.print(rain_rate_per_hour, 2);
            Serial.println(" mm ");
            Serial.print("Rain rate per day =");
            Serial.print(rain_rate_per_day, 2);
            Serial.println(" mm ");
  Serial.println(" ");

  //Sending data to both Google Sheets and SD card every minute
  if (second.equals("0"))
```

```
  {
    checkWiFiStatus();
    sendData(d, t, rain_rate_per_minute, weather);
  }
}

//Function to send data to Google Sheets and SD card
void sendData(String d, String t, float rain_rate_per_minute, String weather)
{
  if (WiFi.status() == WL_CONNECTED)
  {
    //Upload to sheets
    String urlLink = "https://script.google.com/macros/s/" + GAS_ID +
"/exec?" + "&date=" + String(d) + "&time=" + String(t) + "&rain=" +
String(rain_rate_per_minute) + "&weather=" + String(weather);
    Serial.print("POST data to spreadsheet: ");
    Serial.println(urlLink);

    HTTPClient http;
    http.begin(urlLink.c_str());
    http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
    int httpCode = http.GET();
    Serial.print("HTTP Status Code: ");
    Serial.println(httpCode);

    //Response from sheets
    String payload;
    if (httpCode > 0)
    {
      payload = http.getString();
      Serial.println("Payload: " + payload);
    }
    http.end();
```

```
  }

  //Upload to SD Card
  String SDLogging = String(d) + "," + String(t) + "," +
String(rain_rate_per_minute) + "," + String(weather) + "\n" ;

  appendFile(SD, "/rainrate.txt", SDLogging.c_str());
}

//Function to connect to WiFi
void connectToWiFi()
{
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  int wificount = 0;
  while(WiFi.status() != WL_CONNECTED)
  {
    delay(1000);
    Serial.println("Connecting to WiFi...");
    wificount++;
    if (wificount>10)
    {
      ESP.restart();
    }
  }
  Serial.println("Connected to WiFi!");
}

//Function to check WiFi status
void checkWiFiStatus()
{
  if (WiFi.status() != WL_CONNECTED)
```

```
  {
    connectToWiFi();
  }
}

//Function to read SD card file
void readFile(fs::FS &fs, const char * path)
{
  Serial.printf("Reading file: %s\n", path);

  File file = fs.open(path);
  if(!file)
  {
    Serial.println("Failed to open file for reading");
    return;
  }

  Serial.print("Read from file: ");
  while(file.available())
  {
    Serial.write(file.read());
  }
  file.close();
}

//Function to create SD card file (overwrite anything in existing file with
same name)
void writeFile(fs::FS &fs, const char * path, const char * message)
{
  Serial.printf("Writing file: %s\n", path);

  File file = fs.open(path, FILE_WRITE);
  if(!file)
```

```
  {
    Serial.println("Failed to open file for writing");
    return;
  }
  if(file.print(message))
  {
    Serial.println("File written");
  }
  else
  {
    Serial.println("Write failed");
  }
  file.close();
}

//Function to add thing into SD card file (does not overwrite)
void appendFile(fs::FS &fs, const char * path, const char * message)
{
  Serial.printf("Appending to file: %s\n", path);

  File file = fs.open(path, FILE_APPEND);
  if(!file)
  {
    Serial.println("Failed to open file for appending");
    return;
  }
  if(file.print(message))
  {
    Serial.println("Message appended");
  }
  else
  {
    Serial.println("Append failed");
```

```
    }
    file.close();
}
```