Faculty of Electrical and Electronic Engineering Technology

DEVELOPMENT OF IOT BASED HOME WEATHER STATION EQUIPMENT TO MEASURE WIND SPEED AND DIRECTION

MUHAMMAD SYAKIF IEMRAN BIN SABRI

Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics) with Honours

2023

DEVELOPMENT OF IOT BASED HOME WEATHER STATION EQUIPMENT TO MEASURE WIND SPEED AND DIRECTION

MUHAMMAD SYAKIF IEMRAN BIN SABRI

**A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics)
with Honours**

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2023

DECLARATION

I declare that this project report entitled "Development of IoT Based Home Weather Station Equipment to Measure Wind Speed and Direction" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : Syakif

Student Name : MUHAMMAD SYAKIF IEMRAN BIN SABRI

Date : 27 JANUARY 2023

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics) with Honours.

Signature :

Supervisor Name : TS. SULAIMAN BIN SABIKAN

Date : 27 JANUARY 2023

DEDICATION

This study is wholeheartedly dedicated to my beloved parents, who have been my source of inspiration and gave me strength throughout this journey, who continually provide their moral, faith, spiritual, emotional, and financial support.
To my brothers, sister, relatives, mentor, friends and course mate who shared their words of advice and encouragement to finish this study.

ABSTRACT

These days, unpredictable climate change occurs at anytime and anyplace without our consciousness that could affect our lives miserably through natural disaster or severe illness. This is where weather station that equip with IoT come into place where people can monitor real time weather data through smartphone device within WiFi connectivity. These systems are intended to keep an eye on critical occurrences in the physical world, producing information that can be uploaded to the cloud where it can be accessed by applications and from which early precaution can be taken. This project employs 32-bit microcontroller, anemometer sensory part, ESP32 Wi-Fi module where data sends transmissions to cloud service which is Blynk server where it is processed and stored. An application which is Blynk IoT can connect to the intended device to show users the rcorded weather data through smartphone. Therefore, wind vane and 3 cup anemometer has been calibrated against reliable tools to ensure their preciness before installing weather station outdoor for analysis purpose. The wind speed of this weather station has been verified against digital anemometer and has percentage error less than 5%. Wind direction has been verified through digital compass to calibrated its accurateness of direction. The data is collected, analyzed and compared with other weather forecast website like Zoom Earth which is a live weather satellite, Public InfoBanjir, and Malaysian Meteorological Department. These are the most reliable source for weather guidance whether locally or globally for public who keen to know about regional weather status. This product is designed to withstand such blustery weather condition in a long run. Thus, having this personal weather station equipment at household is pretty convenient to monitor real-time weather.

ABSTRAK

Hari-hari ini, perubahan iklim yang tidak dapat diramalkan berlaku pada bila-bila masa dan di mana sahaja tanpa kesedaran kita yang boleh mempengaruhi kehidupan kita dengan teruk melalui bencana alam atau penyakit yang teruk. Di sinilah stesen cuaca yang dilengkapi dengan IoT berada di tempat di mana orang dapat memantau data cuaca masa nyata melalui peranti telefon pintar dalam sambungan WiFi. Sistem ini bertujuan untuk mengawasi kejadian kritikal di dunia fizikal, menghasilkan maklumat yang dapat dimuat ke awan di mana ia dapat diakses oleh aplikasi dan dari mana langkah berjaga-jaga awal dapat diambil. Projek ini menggunakan mikrokontroler 32-bit, bahagian deria anemometer, modul Wi-Fi ESP32 di mana data menghantar penghantaran ke perkhidmatan awan yang merupakan pelayan Blynk di mana ia diproses dan disimpan. Aplikasi yang merupakan Blynk IoT dapat menyambung ke peranti yang dimaksudkan untuk menunjukkan kepada pengguna data cuaca yang diselaraskan melalui telefon pintar. Oleh itu, arah angin dan anemometer 3 cawan telah dikalibrasi terhadap alat yang boleh dipercayai untuk memastikan ketepatannya sebelum memasang stesen cuaca di luar untuk tujuan analisis. Kelajuan angin stesen cuaca ini telah disahkan terhadap anemometer digital dan mempunyai peratusan kesalahan kurang dari 5%. Arah angin telah disahkan melalui kompas digital untuk menentukur ketepatan arahnya. Data dikumpulkan, dianalisis dan dibandingkan dengan laman web ramalan cuaca lain seperti Zoom Earth yang merupakan satelit cuaca langsung, Public InfoBanjir, dan Malaysian Meteorological Department. Ini adalah sumber yang paling dipercayai untuk panduan cuaca sama ada secara tempatan atau global untuk orang ramai yang ingin mengetahui mengenai status cuaca serantau. Produk ini direka untuk menahan keadaan cuaca yang tidak menyenangkan dalam jangka masa panjang. Oleh itu, mempunyai peralatan stesen cuaca peribadi ini di rumah tangga cukup mudah untuk memantau cuaca masa nyata.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Encik Sulaiman Bin Sabikan from the Faculty of Electrical and Electronic Engineering Technology for precious guidance, words of wisdom and patient throughout this project. I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support through completion of this project.

My highest appreciation goes to my parents, and family members for their love and prayer during the period of my study. An honourable mention also goes to fellow colleague and classmates from BEEA cohort 9 for their willingness of sharing his thoughts and ideas regarding the project. I wish nothing but all of success in you all future endeavours.

Lastly, I would like to thank all the staffs at the Faculty of Electrical and Electronic Engineering Technology, the Faculty members, as well as other individuals who are not listed here for being co-operative and helpful. Thank you to everyone who had been to the crucial parts of realization of this project even it’s small, but it’s helpful and meaningful to me.

TABLE OF CONTENTS

PAGE

APPROVAL

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

°C	-	Degree Celcius
mm	-	Millimeter
Ω	-	Ohm
MPH	-	Miles Per Hour
Hz	-	Frequency
m/s	-	Meter Per Second
°	-	Degree
km/s	-	Kilometer Per Second
V	-	Voltage
%	-	Percentage

LIST OF ABBREVIATIONS

USB	-	Universal Serial Bus
Wi-Fi	-	Wireless Fidelity
IO	-	Input / Output
SSID	-	Service Set IDentifier
CPU	-	Central Processing Unit
PCB	-	Printed Circuit Board
I2C	-	Inter-integrated Circuit
HMI	-	Human Machine Interface
GPIO	-	General-purpose Input / Output
ADC	-	Analog to Digital Converter
PWM	-	pulse width modulation
NodeMCU	-	Node MicroController Unit
UART	-	Standard asynchronous serial communication.
Bit	-	Binary Digit
SPI	-	Master / slave communication
DAC	-	Digital to Analog Converter
IP	-	Ingress Protection
IoT	-	Internet of Things
IC	-	Integrated Circuit
VNC	-	Virtual Network Computing
EMC	-	Electromagnetic compatibility

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

People were completely reliant on sensory observations for any weather data until the 18th century. For example, people estimated the time of rain based on the number of clouds in the sky or understood the wind direction based on the movement of leaves. Weather stations have become quite popular nowadays as people can forecast everything regarding the weather quickly and accurately as possible so they can prepare for threatening weather conditions that can occur at any time. The weather station collects meteorological data regarding the current state of the weather in a particular region and makes forecasts regarding the weather conditions in a particular area like prediction about typhoon alarming from media to public as shown in Figure 1.1. Monitoring and alerting of the current weather conditions, including temperature, pressure, humidity, wind speed, wind direction, and the like as they occur in real time that keep on changing rapidly and unpredictably can helped to take initial precautionary.

3 JANUARI 2022
METMALAYSIA
Amaran angin kencang dan laut bergelora, dijangka landa beberapa negeri sehingga esok
BULETINTV3
www.buletintv3.my
Buletin TV3

Figure 1.1 News about typhoon alarming from Malaysian Meteorological Department

So, it is essential to keep a record of short-term weather updates in order to acquire accurate information regarding the state of the weather. In this case, the weather station plays an important mechanical method and having a personal weather station is necessary for household in modern day. Extensive data about the wind direction and speed for example is collected from weather stations across the world and made available online. Knowing the direction of the wind blowing is important since it provides information about storm systems that can expect soon. But with a quick glance, these simple instruments which is weather station that can be place at household can give first-hand knowledge of wind direction and wind speed. The use of weather stations that can observe the real-time of wind condition can helps to cope and can informed people well in advance about the prevailing weather condition with the Monsoon wind sign that might cause climate disaster in Malaysia such as flood or typhon. This project can measure real-time wind speed and direction at installed area using low-cost high precision calibrated anemometer for accurate measurement of parameter. This recorded data is then sent to the cloud platform for storage, processing, monitoring, notification, and prediction.

1.2 Problem Statement

Recent days, we can find out there are several of home weather station that can be purchased in the marketplace that can measure various of parameters such as rain fall, temperature, humidity, Barometric pressure, and wind parameter. However, there are still lack of invention for those weather station such as high cost to purchase, hard to setup and lack of connectivity range that can only be monitoring in short range area of installed weather station. Any method that is used to forecast the weather must adhere to the standards of accuracy and dependability. It also needs to make it simple to access all of the parameters that are being measured. Some people were preoccupied with domestic activities or office work and had no idea about the natural boundaries and weather variations beyond their home or office, especially those who travelled frequently. A location's weather conditions are determined by several factors, the like of wind speed and direction so stormy climate can be predicted. This equipment can provide an effective solution in these circumstances and dependable for monitoring the weather or climate conditions in a specific area using the Internet of Things (IoT) capability, which has the capacity to monitor and observe the wind condition anywhere from a remote area where does weather station is placed by utilizing the internet network. Keeping weather station able to withstand in its own place even in blustery conditions is also important circumstances as well to prevent electronics part from get fail or start to corrode.

1.3 Project Objective

The following is a list of the objectives that this study aims to achieve:

a) Design a based home weather station equipment that take precise readings of wind speed and direction in real time using active mechanical devices that physically interact with their surroundings.

b) To develop weather station that equip with IoT so it can be observe and monitor its data from smartphone within internet connectivity.

c) To develop portability usage, robust, weatherproof and reliable prototype based home weather station where its components implement in PCB board layout.

d) To analyze the real-time data from weather station.

1.4 Scope of Project

The information regarding the parameters that were measured is not helpful unless it is transmitted to the users promptly and correctly. Therefore, having a IoT based home weather station can help instantly provide alerts regarding abnormal changes in the weather on user through phone. Therefore, user can easily prevent any mishap or loss due to their forecasting and instant alerts. Also, user can share wind parameter data on various devices by connecting the weather station to a Wi-Fi network. The processing and transmission of measured data is a critical component of modern weather forecasting. The Internet of Things (IoT) is being implemented in this project so this project's application will involve monitoring the meteorological wind parameters. Thus, the scope of this study is as follow:

a) The design of the project is balanced for all these factors in terms of reliability, ease of use, accuracy, and cost.

b) The app interface provides user-friendly interface for monitoring weather parameter data.

c) The prototype is weatherproof and able to hold all of the components in place to measure precise weather parameters data.

d) The system required consistency of wind flows intensity at least to obtain best performance.

1.5 Report Structure and Organization

This thesis has five chapter that are outlined as follows:

Chapter 1 explained the background of weather station. Based on the problem statement that faced, few objectives are established as a guidance for further reference

In Chapter 2, literature study is conducted to provide knowledge while understanding of the other researches to reduce unresolved problems and prevent repetition of project occurs. The comparison between previous researches works are done to conclude the advantages and disadvantages of their methods.

In Chapter 3, methodology is formulated into four milestones that referred to the objectives for this project. The system architecture able to be designed based on the previous research studies by selecting the right methods and the components that included in this project. Testing and troubleshooting is evaluated based on the algorithm development in this system. Furthermore, data collection and analysis are analyzed to ensure this system achieved the objectives stated in this project. Finally, the prototype of the system is developed to fit all components.

Chapter 4 shows the results and analysis that should achieved by the system designed after the basic testing for the components. This result is used as a guidance for further works in second task of the final year project.

Lastly, Chapter 5 conclude all the efforts and works inside this project and a short summary for all the process that have been done in this project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This section discusses and summarize the overall concept and theory behind the project, which is based on a home weather station that uses equipment to measure wind speed and direction. The primary argument of this chapter was expounded upon using the research both from the past and the present. In this chapter, it talked about the concept and theory that were used to solve the problem with this project. Journals, articles, and case studies make up the bulk of the information that was gathered, and these formats were chosen for their similarity to the project's scope in order to ensure accuracy.

2.2 Current Wind Speed and Direction Equipment

Personal weather station, also known as home weather station can perform tasks that are not possible with smartphone apps. Equip with sensory part, it can monitor exactly how the weather’s doing in real-time by measuring atmospheric conditions, in this project is mainly focus on wind speed and direction.

2.2.1 Digital Anemometer

One variety of anemometer is called a digital anemometer shown as Figure 2.1, and it measures both the speed of the wind and the volumetric flow. In most cases, they take the form of compact handheld devices that combine a fan and an electronic display. It is possible for the turbine (which is frequently referred to as a propeller, impeller, or fan) to be either built into the device itself or provided as a separate sensor plugin. This kind of anemometer is very portable and easy to carry, but it cannot be used to measure wind direction because there is no wind vane mechanical structure. Wind vane is instrument that show which way the wind is blowing, so without them, the anemometer would be useless to measure any wind direction. In addition, the range of wind speed that can be measured is quite limited, and the data cannot be viewed remotely.

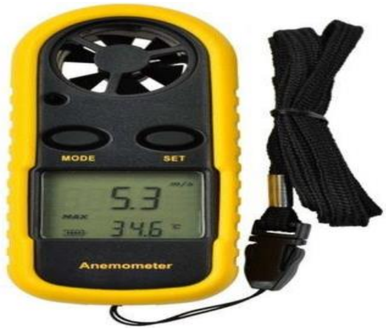


Figure 2.1 Digital Anemometer

2.2.2 Windsock

A windsock is a textile tube in the shape of a cone that looks like a giant sock Figure 2.2. The direction and velocity of the wind can be roughly determined by looking at windsocks. Windsocks are illuminated at night either by floodlights mounted on top surrounding it or by one mounted on the pole shining inside it. A windsock that is pointing due north indicates that the wind is blowing from the south, because the wind direction is the opposite of the direction that the windsock is pointing. The angle of the windsock in relation to the mounting pole is a good indicator of the speed of the wind. When there is little wind, the windsock hangs down, and when there is a lot of wind, it flies horizontally. In the beginning, a method that assisted in determining the speed of the wind was to use alternating stripes of high visibility orange and white. The estimated wind speed is increased by three knots for every stripe that is present. However, the mountings of some circle frames cause windsocks to be held open at one end, which indicates a velocity of three knots even though anemometers would not show any wind speed.

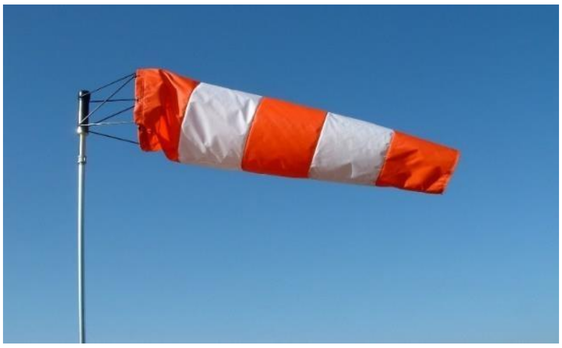


Figure 2.2 Windsock

2.2.3 Traditional Wind Vane

One of the earliest meteorological instruments is the wind vane shown as Figure 2.3, which shows the direction of the wind blowing. A wind vane is a device that measures the direction of the wind's movement. An elevated shaft or spire is used to mount a wind vane. The rudder blade is turned by the wind. The direction of the wind is indicated by the vane's narrow end, which points into the wind.

Modern wind vanes are highly technical instruments with extremely high sensitivity levels. At low wind speeds, they are extremely accurate and have a low starting point for turning. Even the tiniest shift in wind direction can be picked up by this device. Today, they are built to withstand stormy weather for an extended period of time.

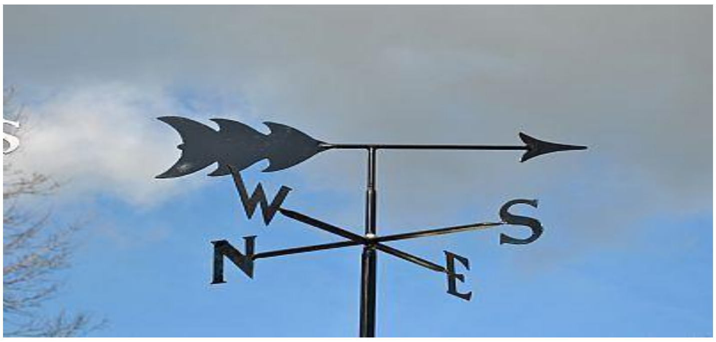


Figure 2.3 Wind Vane

2.2.4 Ultrasonic Anemometer

Airflow speed and the static sound wave propagation speed are equal to each other in determining the actual sound wave propagation speed. This means that sound waves propagate more slowly downwind than upwind after a certain distance. The ultrasonic anemometer shown as Figure 2.4 can measure air velocity by measuring this time difference.

Wind speed and direction can be measured using an ultrasonic anemometer. Mechanical anemometers have inherent flaws that make them ineffective in all weather and for a long period of time, so the use of electronic anemometers is on the rise. Mechanical anemometers will no longer be needed thanks to this new technology. The setup is more difficult, and the cost is significantly higher.

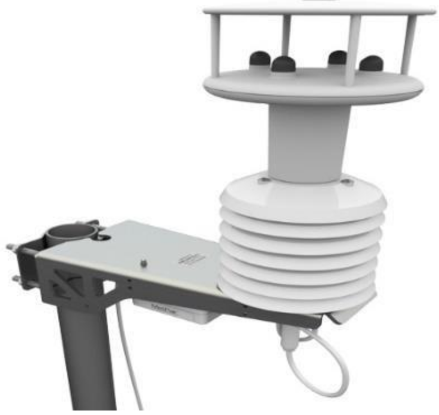


Figure 2.4 Ultrasonic Anemometer

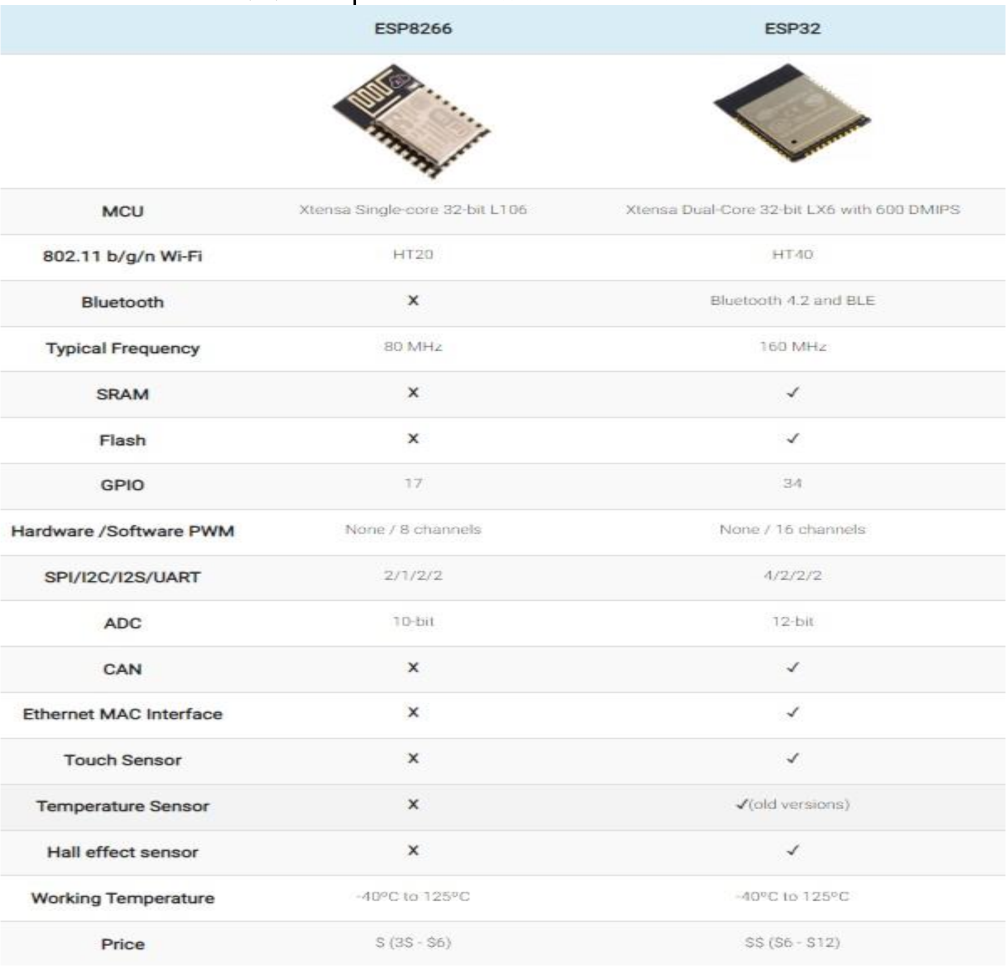
2.3 Studies Related to Home Weather Station's Components

In this section, the components that used widely among researchers are well explained. Those specification of components are referred on the datasheet of the component to avoid wrong information.

2.3.1 Wi-Fi Module

There are two Wi-Fi modules that suite with Arduino IDE software which is the ESP32 and ESP8266 that are quite affordable to be use. Both of this Wi-Fi modules perfectly suited for this project that require the use of the Internet of Things (IoT). A 32-bit processor is found in both chips. The ESP8266 has a single core and runs at 80MHz, whereas the ESP32 has a dual-core CPU running at 160MHz to 240MHz. SPI, I2C, UART, ADC, DAC, and PWM are just a few of the protocols supported by these modules' GPIOs. To top it all off, unlike other microcontrollers like the Arduino, these boards include built-in wireless networking. Thus, wireless or Bluetooth remote control and monitoring of devices is now possible (in the case of ESP32).

Table 2.1: Comparison between ESP8266 and ESP32



	ESP8266	ESP32
MCU	Xtensa Single-core 32-bit L106	Xtensa Dual-Core 32-bit LX6 with 600 DMIPS
802.11 b/g/n Wi-Fi	HT20	HT40
Bluetooth	✗	Bluetooth 4.2 and BLE
Typical Frequency	80 MHz	160 MHz
SRAM	✗	✓
Flash	✗	✓
GPIO	17	34
Hardware /Software PWM	None / 8 channels	None / 16 channels
SPI/I2C/I2S/UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	✗	✓
Ethernet MAC Interface	✗	✓
Touch Sensor	✗	✓
Temperature Sensor	✗	✓(old versions)
Hall effect sensor	✗	✓
Working Temperature	-40°C to 125°C	-40°C to 125°C
Price	$ ($3 - $6)	$$ ($6 - $12)

Table 2.1 above show adirect comparison between ESP8266 and ESP32. ESP32 is identified as the most recent Wi-Fi module technology because of its extra features to communicate with each of the sensors and allows for an external internet connection to a Blynk server to push collected data out.

2.3.2 IC Voltage Regulators

This project calls for a voltage regulator component that can manage the 5V source voltage from the USB adapter, as most 32-bit microcontrollers need a clean and reliable 3.3V or 1.8V supply to run. A constant DC output voltage from a three-terminal IC voltage regulator that is unaffected by changes in input voltage, output load current, or operating temperature. Linear IC voltage regulators, switching IC voltage regulators, and DC/DC converter chips are the three types of IC voltage regulators. In order to convert the input voltage into a consistent output voltage, IC linear voltage regulators employ an active pass element. In contrast, integrated circuit switching voltage regulators (ICs) store energy in an inductor, transformer, or capacitor and transmit it in discrete packets from the input to the output through a low-resistance switch. A third kind of IC voltage regulators, DC/DC converter chips, take an unregulated input voltage and produce a regulated DC voltage. Furthermore, DC/DC converters isolate and regulate power buses from noise. Each IC voltage regulator has an adjustable or set output voltage that falls within a narrow range. Multiple options for functionality can be found in IC voltage regulators. The ability to have a number of different channels or outputs, Reverse voltage protection prevents damage in applications where users can accidentally reverse battery polarity, thermal shutdown protection turns off IC voltage regulators when the temperature reaches a predefined limit, and shutdown (inhibit) pins are used to disable regulator outputs. The block diagram of a three-terminal IC voltage regulator is shown in Figure 2.5.

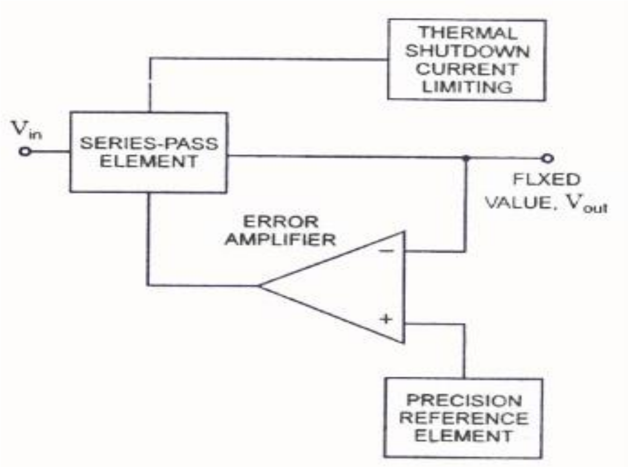


Figure 2.5 Block diagram of a three-terminal IC voltage regulator

2.3.3 32-Bit Microcontroller

A microcontroller is a component of a semiconductor chip that performs arithmetic processing and controls the circuit via the I/O and peripheral interface. The designation '32-bit microcontroller' implies that the microcontroller can perform arithmetic operations on a 32-bit number. Due to its larger data bus, the 32-bit microcontroller requires less instruction cycles to execute a function than an 8-bit microcontroller. Because of its higher performance, a 32-bit microcontroller is frequently constructed with extra peripherals and memory. Because 32K is greater than both 8 and 16, 32-bit microcontrollers can transport more data in a given time period than 8-bit and 16-bit microcontrollers. As a result, a 32-bit microcontroller can manage triple the amount of data as an 8-bit or 16-bit CPU, making the 32-bit microcontroller more data efficient. A 32-bit microcontroller can handle many peripherals efficiently, but it consumes more power, especially when all embedded systems and peripherals are turned on. A 32-bit microcontroller is used for any application that demands computations that necessarily involve huge numbers and must be performed quickly. A weather station system that demands massive data processing of its parameters and offers a response in a fraction of a second. second.

Table 2.2: Difference between 8-bit, 16-bit and 32-bit microcontroller

	8-bit microcontroller	16-bit microcontroller	32-bit microcontroller
Data bus	8-bit	16-bit	32-bit
Physical memory	1KB	1MB	4GB
Data range	255	65,535	4,294,967,295
Clock speed	8MHz	40-64MHz	Over 100MHz
Virtual addressing	Not support	Not support	64TB virtual memory
Form Factor	Dip packages (Arduino boards)	Dip packages, MLF, TQFP, and QFP	TQFP, QFP, VTLA, TFBGA and LQFP

Table 2.2 shows difference between 8-bit, 16-bit and 32-bit microcontroller in general. There also have several different package types of microcontrollers in current market these days as shown in Figure 2.6.

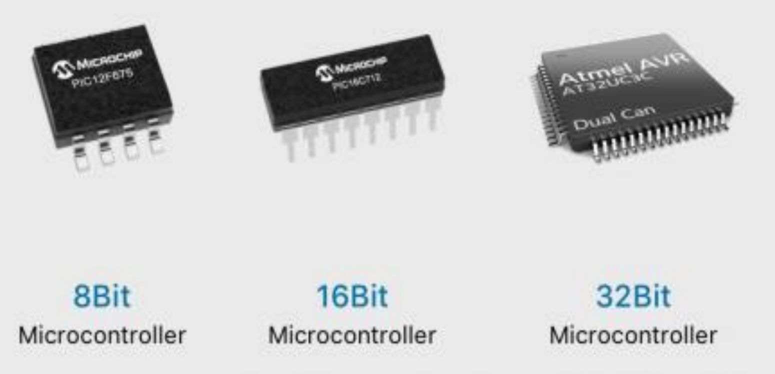


Figure 2.6 Microcontroller package types

2.3.4 Wind Vane

A wind vane is an instrument that measures the direction of the wind. They were commonly seen on top of buildings along the coast or aboard ships. Many were highly decorative. A wind vane is now an integral component of a weather station. A wind vane is installed on a shaft or spire that is elevated above the ground. The wind catches the rudder blade and causes it to rotate. As seen in Figure 2.7, the narrow end of the vane points into the wind and indicates the direction the wind is coming from. Wind vanes are extremely technological and sensitive instruments. They feature good accuracy in low wind speeds and a low turning threshold. This implies that it can detect even little changes in wind direction. They are now designed to be long-lasting and able to resist stormy weather. Wind cups are now attached to modern ones to measure wind speed. They share the same axis and can provide a coordinated readout of both pieces of information. It is recommended to mount a weather vane for non-agricultural use at a height of about 30 feet above the ground, preferably over level terrain. When mounting the device, the distance between the wind sensors and the nearest major object should be twice the object's height. Otherwise, it will be detecting wind turbulence caused by the object rather than ambient wind. Although perfect placement is not always achievable, well-designed wind vanes and wind cups will provide accurate data regardless of where they are placed. Wind vane data can be used to calculate average wind direction and fluctuations over certain time periods. Meteorologists need to know which way the wind is blowing. It provides information about storm systems and what to expect in the near future. Wind direction data can aid in determining the location of a pressure center. Wind direction is significant in determining the direction of wildfire flames and smoke, as well as hazardous substance spills. Mariners and sailboat operators require wind direction information as well. They are vital to the aviation industry in a variety of ways. It is particularly vital for those working in the construction business, especially when working on high-rise buildings that use cranes.

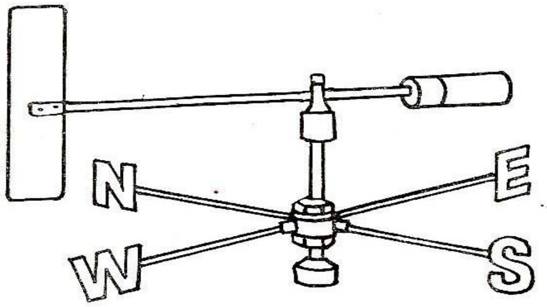


Figure 2.7 Wind Vane mechanism

2.3.5 3 Cup Anemometer

An anemometer is a device that utilize 3 wind cup rotation to determine the speed of the wind shown as Figure 2.8. An anemometer with a rotating wheel design is common model in current days. The faster the wheel spins, the higher the wind-speed readings will be at higher altitudes. The anemometer measures wind speed by counting the number of rotations. Anemometers measure wind speed in mph, kph, m/s, or knots, depending on the model. Three hemispherical cups were mounted on horizontal arms and a vertical shaft to make up the cup-type anemometer. Shaft rotation was inversely proportional to wind speed when cups were placed horizontally in the wind. As a result, for a wide range of wind speeds, counting the shaft's turns and produced a value tally to the accurate wind speed. Meteorologists use anemometers to analyze weather patterns and calculate wind pressure. Structures are subjected to the wind's pressure because of the force it exerts on them. a change in weather patterns, such as an approaching storm, can be predicted by monitoring the speed of the wind.

The amount of wind pressure on a surface, such as a cup or a propeller, or the use of sonic pulses are two ways anemometers gauge wind speed. With a wheel and cups or a propeller at the ends of the spokes, an anemometer works. A magnet can be found in one of them. It records every time the magnet touches a switch. The wind speed can be accurately measured with this device. Cup or propeller anemometers, for example, count the revolutions per minute to obtain an electronic reading of the wind speed. Wind speeds may be higher at higher elevations. The angle of the vane and the minimum air velocity required to rotate the vane can both affect the accuracy of the readings. Elevation, adjacent landforms such as valleys or mountains, and trees or buildings that may obstruct wind are all factors that might influence wind source. Placing anemometers near mountains, valleys, or canyons can improve wind speed reading accuracy.

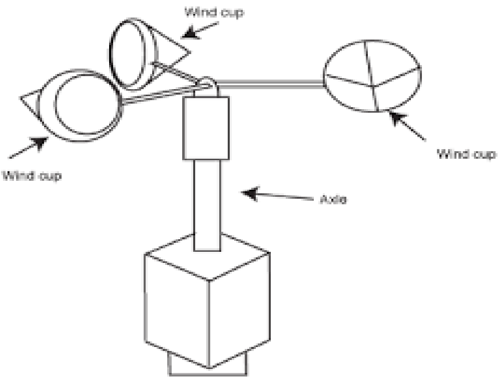


Figure 2.8 3 Cup Anemometer configuration

2.4 Previous Related Research Work

The previous research or articles are revised before further development in home weather station equipment. The main purpose of readings previous research work is to discuss the theory and concept that used to solve the problems that will be faced during the development in this project. These sources have been chosen that related to the objective for this project.

2.4.1 A Weather Station Design Using IoT Platform Based On Arduino Mega

The purpose of constructing a weather station is to gather numerical data about a location's weather conditions. Because of the day-to-day fluctuations in the weather, it's critical to keep an eye on the weather in any given location. Through the use of an IoT platform, this study aims to create a weather station that can be accessed via a website. The weather in an area can be learned about without having to travel to the area. The microcontroller in this design is an Arduino Mega 2560 [8]. A temperature sensor, humidity sensor, rain sensor, and an air pressure sensor (BMP180) were used to collect data on the weather. Weather forecasts are based on air pressure measurements. A TFT LCD 2.2 and an ESP8266 Wi-Fi module are used to display the data collected by each sensor, and the results are saved to an SD Card. The PCE-THB 40 module used in this study had an average measurement error of 3.74% for temperature, 2.14% for air humidity, and 0.32% for air pressure [4].

2.4.2 Power Management In IoT Weather Station

The researcher demonstrates a battery-powered Internet of Things (IoT) weather station prototype with a power management solution. This system was tested with real weather data using a complete simulation software. An optimized system runtime was calculated and implemented with the help of the developed software and power consumption measurements made on the prototype [3]. The calculations were fine-tuned to allow the switch from batteries to supercapacitors in order to reduce maintenance.

2.4.3 Efficient IoT based Weather Station

A new method for making practical and meaningful use of technology in a smart weather station system is described herein. The efficiency of a weather station is enhanced by the instruments and equipment it uses to collect atmospheric data on the expected weather conditions. Weather monitoring is essential because the weather can change at any given moment. Take a look at the latest weather reports to get a sense of the situation on the ground. As a result, we can plan our activities in accordance with the weather. The effects of the weather on human physical and psychological well-being cannot be overstated. As a result, we must always be aware of the most recent weather conditions. This is where a weather station comes in handy, as it keeps us informed of the current weather conditions. Using apps, we can quickly and easily see weather updates and data from weather stations. Using the NodeMCU Board and Blynk IoT technology, this paper has developed and tested a weather station that can measure a wide range of meteorological parameters such as temperature, pressure, humidity, and rainfall [1].

2.4.4 Cloud Based Weather Station using IoT Devices

The smart system cloud-based weather station is the subject of this study by researcher. The weather data is collected and analyzed using a Raspberry Pi. Cloud-based weather data storage and processing is used to predict the impact of this weather change. Weather parameters like temperature, humidity, wind speed and pressure, and rainfall can all be monitored by the system. System design goals include making it simple, low-cost, and low maintenance for users to interact with it. The system is built using Raspberry Pi, various sensors, and Wi-Fi as a communication medium, which consumes very little power and is very inexpensive to build. The sensor data is sent to the base station Raspberry Pi. Once the data has been transferred to the cloud database, it is used to train a new machine learning model and to monitor and study various weather patterns and trends using the Raspberry Pi 3 [2]. Through a cloud-based web application built with Django Framework, weather data and insights can be accessed by remote users.

2.4.5 Design and Implement of a Weather Monitoring Station using CoAP on NB- s IoT Network

Prototyping a weather station for the purpose of collecting and monitoring meteorological data is at the heart of this project's investigation. Temperature, humidity, wind speed, wind direction, ozone gas and atmospheric pressure can all be measured by the weather station's Arduino board and other devices. Device diversity, low cost, long battery life, and high density of connections were all key considerations in the design of this system. As a result, data will be transferred to the MySQL database server using the Narrowband Internet of Things Network (NB-IoT) (CoAP) [10]. Grafana (Open-source visualization and analytics software) can be used on a personal computer to display received data. For those who rely on weather data in their daily routines, this system will be a huge help.

2.4.6 An IoT-based Smart Garden with Weather Station System

Connected devices that can communicate with each other are known as the Internet of Things (IoT). Data can be exchanged between these devices and the user. IoT-enabled Smart Gardens with Weather Stations can be used to monitor plant growth every day and predict the likelihood of rain [7]. Despite the fact that many people are interested in gardening, they often neglect to water their plants. Because of this, a smartphone is used to monitor and control the device's water pump in this research. Barometric pressure, DHT11 temperature and humidity sensor, soil moisture sensor, and light intensity module sensor are the four main sensors in the devices. Percentage measurements were made using the Soil and Light Intensity sensor. Two additional actuators can be used remotely or by pressing buttons on the devices: the water pump and the LED light. In order to speed up the plant's growth, the LED has been designed to mimic sunlight. A smartphone app called Blynk can be used to record and send data from this IoT-based Smart Garden with a Weather Station System to the user. Researchers and farmers, as well as children, can easily use the system to carry out their work [6].

2.5 Summary

Based on the previous researcher works and the theories, theses information that related to this project gained knowledge about the methods used by the previous researchers. Therefore, comparison of methods applied, advantages and disadvantages are made to show the similarities and differences methods applied by previous researchers. In addition, the theories for these components are explained to provide a clear image about the function of each component. Lastly, this based home weather station equipment designed to give accurate and precise weather parameters in real-time to user so they able to monitor weather condition in their area through smartphone over the Wi-Fi connectivity.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The methodology is the systematic, theoretical analysis of the methods applied to this project. In this chapter, there will be 2 parts. It is hardware and software part. This project implementation is the design for wind direction and speed monitoring which connected to 32-bit microcontroller, ESP32 module and for the software implementation, there is programming that will be for the Arduino IDE software and the simulation using Proteus.

The project flow was discussed in depth for this methodology part. In this chapter, a portion of the material was explained the methods used throughout the duration of the project to carry it out. The purpose of this chapter is to provide additional information and confirmation of the way this project was carried out. Development of the design of based home weather station involves the implement of bothhardware and software. This approach had been properly put in place in order to the suitable mechanism and component for this based home weather station equipment.

3.2 Project Milestone

The project milestones are defined before starting the research methodology. The function of listed out those milestones shows the forward progress in this project plan. Milestones also provide technique that can estimate the time it takes to complete this project, making a detailed project with strategy and scheduled. Figure 3.1 shows project milestones.

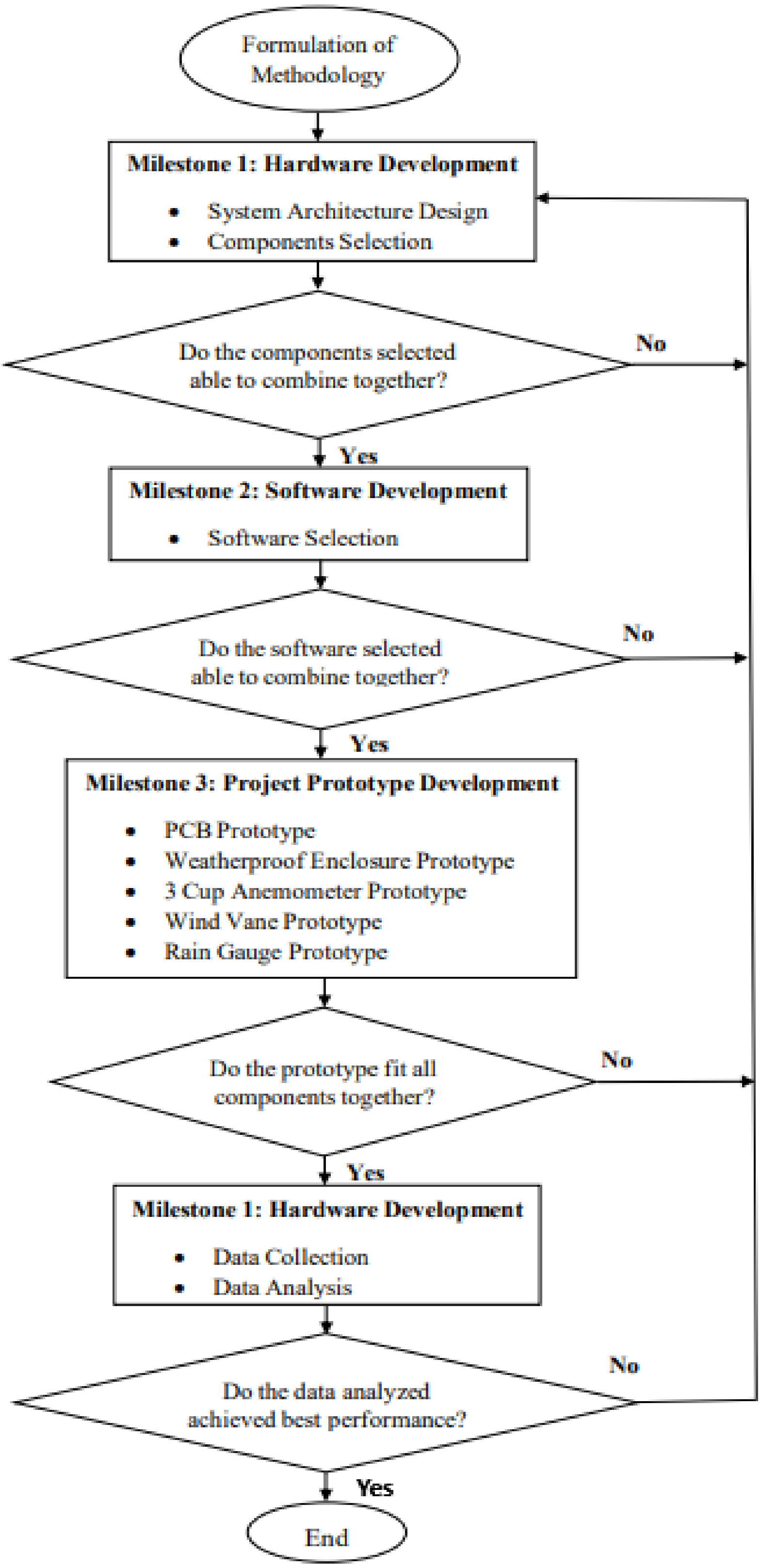

Formulation of Methodology
Milestone 1: Hardware Development
• System Architecture Design
• Components Selection
Do the components selected able to combine together?
No
Yes
Milestone 2: Software Development
• Software Selection
Do the software selected able to combine together?
No
Yes
Milestone 3: Project Prototype Development
• PCB Prototype
• Weatherproof Enclosure Prototype
• 3 Cup Anemometer Prototype
• Wind Vane Prototype
• Rain Gauge Prototype
Do the prototype fit all components together?
No
Yes
Milestone 1: Hardware Development
• Data Collection
• Data Analysis
Do the data analyzed achieved best performance?
No
Yes
End

Figure 3.1 Flowchart for project

3.3 Milestone 1: Hardware Development

This section will be elaborate on choosing the right components that will be used to complete this project. This section also will be elaborate about idea, design, and function of the component that related to the project. Figure 3.2 shows flowchart for milestone 1.

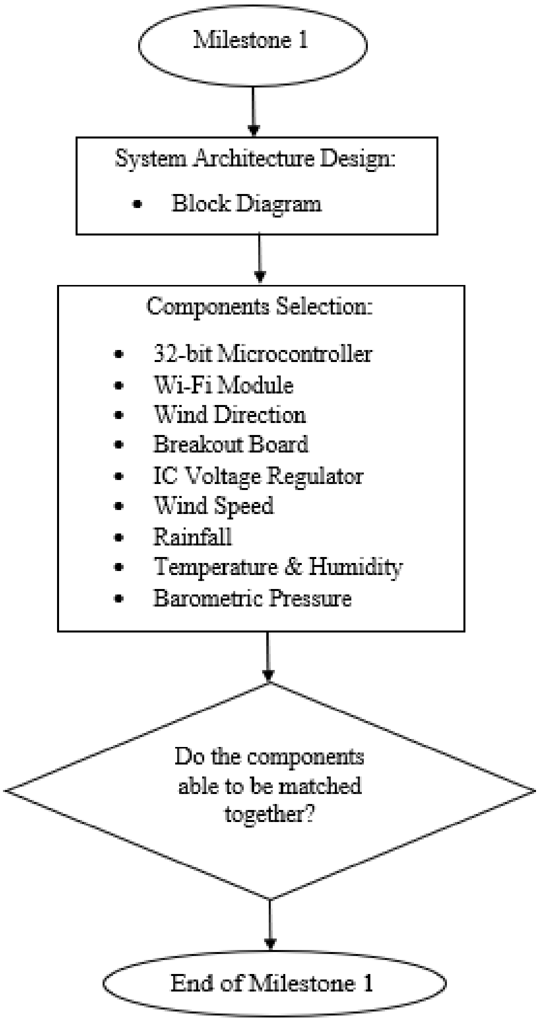


Figure 3.2 Flowchart for Milestone 1

3.3.1 System Architecture Design

The system weather station consists of few parts of components that required the suitable connection between those components, such as anemometer that link with RJ11 connector that isn't compatible with microcontroller connection. In this situation, it needs converter to make it able interface with ESP32 port. Since this system tool is portable, which required adapter as power supply, functionality of the system is always ready as long as there is a source to power the system and reliable internet network. Then, Wi-Fi module ESP32 is connected directly to the microcontroller's IO pin. The system diagram is arranged with the block diagram method as shown in Figure 3.3.

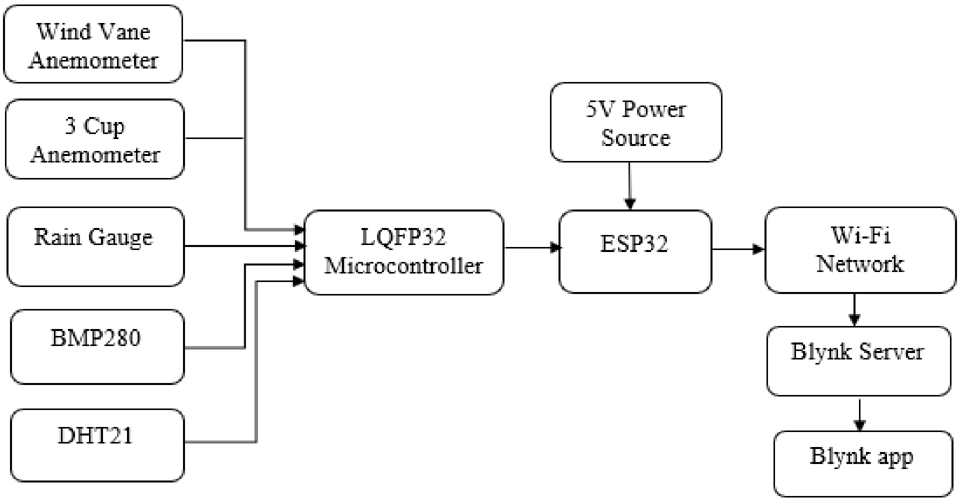


Figure 3.3 Block Diagram for the System

3.3.2 LQFP32 32-bit Microcontroller

Based on the research microcontroller, LQFP32 32-bit Microcontroller shown as Figure 3.4, which a device that has integrated several functions which save job load of the microcontroller and save necessary connections to the microcontroller. It is possible to attach analogue and digital high side drivers (HSD) to the device and control them over the SPI interface. A synchronous detailed diagnostics function is included. The device has 8 outputs to the HSD that can be operated either by steady state ON/OFF mode or by PWM. Two clock inputs are supplied for usage as the base frequency to generate the PWM signal internally. The outputs are entirely independent and can also be operated with phase shift to improve power net characteristics during the inrush phase. To perform diagnostics, the device has 8 current sense (CS)/status (ST) pins attached to the HSD. The index of the ST/CS pin corresponds to the input linked to the same HSD channel. To connect with a microcontroller, this device employs a 16-bit SPI slave protocol constructed according to the ST SPI Standard. Cortex-M3 offers outstanding 32-bit performance with minimal dynamic power. It also provides superior system energy efficiency thanks to integrated software-controlled sleep modes, robust clock gating, and configurable state preservation. This microcontroller is a family of 32-bit microcontroller integrated circuits with the main board itself making it convenient to link it with other components.

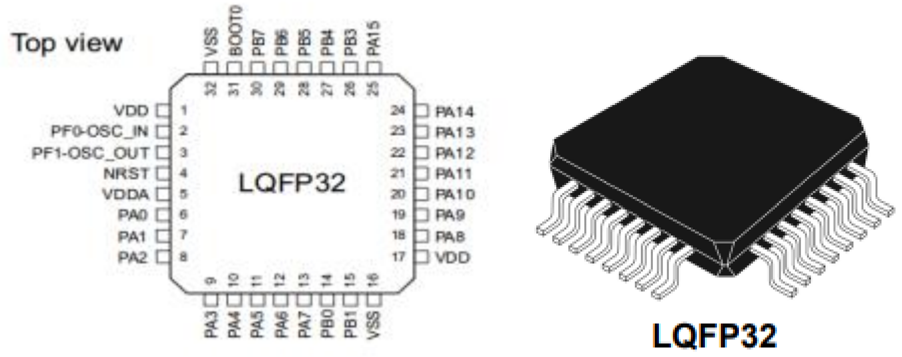


Figure 3.4 LQFP32 package pinout

3.3.3 RJ11 breakout board

Since the anemometer sensor is in RJ11 connector, The basic board connects all six conductors to a breadboard-compatible header is needed as shown in Figure 3.5. This board is perfect for several weather meters, and it can easily connect these sensors to breadboards. If required, a small piece of breakaway header can be soldered onto each breakout board, which includes the RJ11 connector. RJ11 cables that need to be terminated can be quite hard to work with. On their own, the pins are extremely difficult to make contact with, and the mating connectors are not breadboard compatible. To facilitate prototyping and quick connection, this breakout board exposes all six pins of a modular jack. This breakout board offer connections via 0.1 header pins. The RJ11 breakout board incorporates additional male headers strip and feature four mounting holes. I2C (Integrated to Circuit) communication protocol bus, analogue ports, and external interrupts link the sensor to the microcontroller, while VNC (Virtual Network Computing) connects the CPU to the mobile app via the terminal. At last, the data is uploaded through Wi-Fi to a remote server. Tools, user profiles, and data accessibility all work together to form the network layer. It breaks out all 6 pins from the RJ11 connector so make it compatible with some Weather Meters.

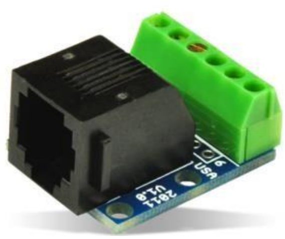


Figure 3.5 RJ11 breakout board

3.3.4 AMS1117

AMS1117 structure as shown in Figure 3.6, is a common 3-pin SMD package voltage regulator that comes in a wide variety of variants to meet both fixed and adjustable voltage needs. Maximum output current from the IC is 1A, and the output voltage is adjustable between 1.5V and 5V. When running at full current, its drop off voltage is only 1.3V. The output voltage can be adjusted from 1.25V to 12V using just two external resistors; the fixed version is available in the following values: Vout = 1.2V, 1.8V, 2.5V, 2.85V,3.3V, and 5V. The AMS1117's thermal shutdown feature helps maintain reliable operation of the chip and power supply. Also, it employs a trimming method that ensures a 2% margin of error in output voltage. On demand, we can also adjust the precision of the voltage we output to within 1%..

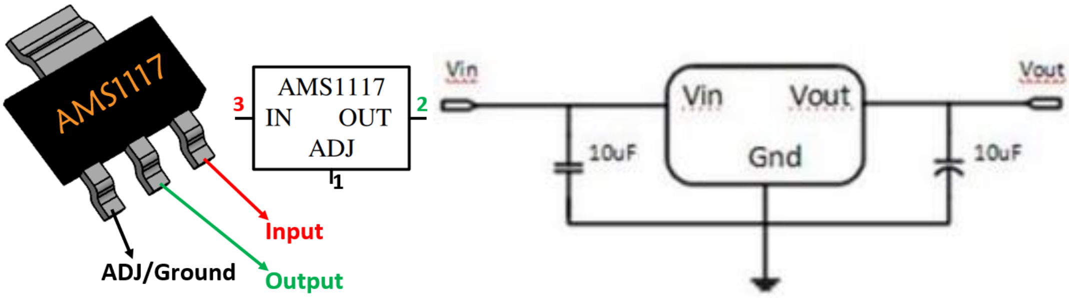


Figure 3.6 AMS1117 3 Pin Voltage Regulator

3.3.5 TTGO T-Display ESP3 2 Development Board

Since this project mainly focusing on how to make the project that require internet connectivity support, ESP32 is the perfect choice. ESP32 is the upgraded model of the ESP8266 module and is designed by Espressif Systems. The ESP32 is a single 2.4 GHz Wi-Fi and Bluetooth combination chip built on TSMC's low-power 40 nm technology. It is intended to provide the best power and RF performance while demonstrating resilience, versatility, and dependability across a wide range of applications and power scenarios. With less than ten external components, ESP32 is the industry's most integrated solution for Wi-Fi + Bluetooth applications. The antenna switch, RF balun, power amplifier, low noise receiving amplifier, filters, and power management modules are all integrated within the ESP32. As a result, the overall system takes up very little Printed Circuit Board (PCB) space. ESP32 employs CMOS for its single-chip fully integrated radio and baseband, as well as innovative calibration circuitries that enable the solution to dynamically adjust itself to remove external circuit flaws or to adapt to changes in external conditions.

The ESP32 is intended for mobile, wearable, and Internet of Things (IoT) applications. It includes numerous characteristics found in cutting-edge low-power chips, such as fine resolution clock gating, power modes, and dynamic power scaling. In a low-power IoT sensor hub application scenario, for example, the ESP32 is woken up periodically and only when a defined condition is detected, with a low duty cycle employed to decrease the amount of energy expended by the chip. The power amplifier's output power can also be adjusted to establish an appropriate balance between communication range, data rate, and power consumption. This module offers sufficient on-board processing and storage capability to connect with sensors and other application-specific devices via its GPIOs while requiring little development and loading during runtime. Because of its high level of on-chip integration, it requires less external circuitry, including the front-end module, which is designed to take up as little PCB space as possible. There is a seemingly unending source of material accessible for the ESP32, thanks to amazing community support. The TTGO T-Display, depicted in Figure 3.7, is an Arduino Development Board powered by an ESP32. It's a portable, user-friendly, open-source IoT development board. This module includes a 1.14-inch LCD with a resolution of 135X240 per SPI protocol, as well as a USB type C port for programming.

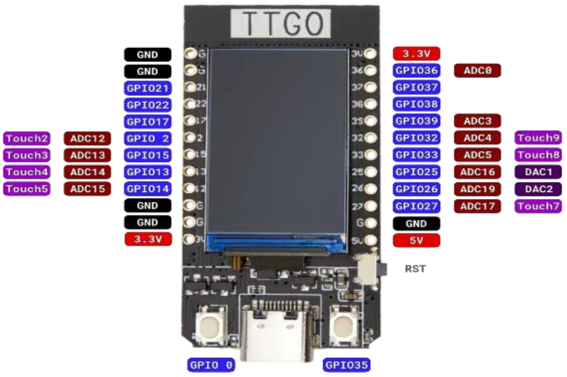


Figure 3.7 TTGO T-Display ESP32 Pin

3.3.6 Wind Anemometer

This anemometer has three arms with scoops on the end that catch the wind and cause the arms to spin. Upon dismantle one of the anemometers as shown in Figure 3.8, a small magnet will be found attached to the underside of the cup. The magnet activates a reed switch, a clever piece of electronics, at two different places in its rotation. When subjected to a magnetic field, the reed switch's internal metal contacts make contact with one another. When the anemometer spins, the magnet briefly forms a closed circuit through the reed switch, acting just like a button attached to the microcontroller. Therefore, the rate of rotation of the anemometer cup can be determined by counting the signals from the reed switch. Reed switches generate a signal that may be picked up by an ESP32's general-purpose input/output (GPIO) pin whenever they are actuated. At least two perceptible signals will be generated by the sensor for every full rotation of the anemometer. These signals can be measured in terms of their duration and frequency, allowing for the determination of wind velocity. Both wires should be connected to GPIO 5, with one going to ground. When connected to an RJ11 connector, the anemometer makes use of the third and fourth pins on the board. These can b directly send the measured signal to microcontroller to be processed and send to ESP32.

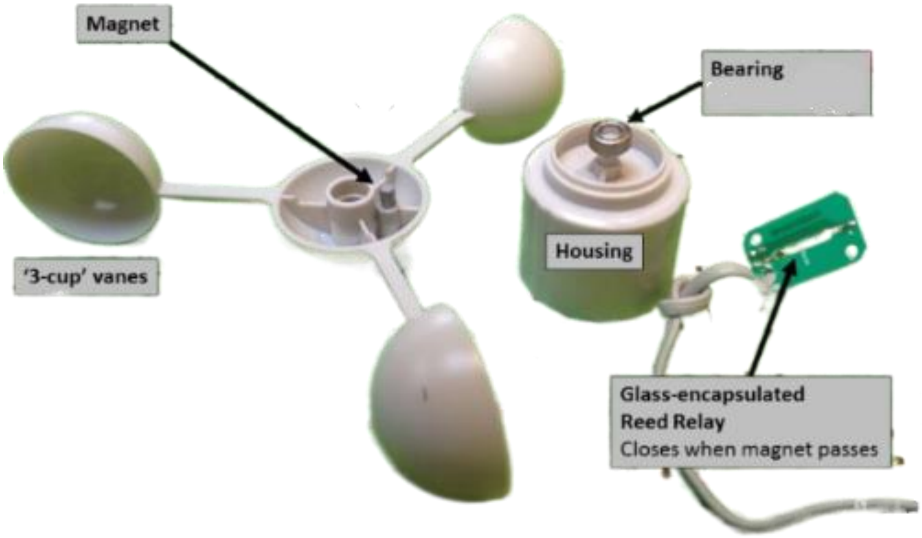


Figure 3.8 Disassembly 3 cup anemometer component

3.3.7 Wind Vane

Wind vanes indicate the direction the wind is blowing from rather than the direction it will go. At first, this may seem counterintuitive, given that the arrows on, say, a TV weather map, point in the opposite direction. When wind blows, it creates a force on the vertical blade of a wind vane, which causes the blade to rotate until it reaches the point of least wind resistance, which coincides with the direction of the oncoming wind. Wind vane used here also has reed switches and a rotating magnet, but it is more complex and works in a completely different way. inside the wind vane, there are eight reed switches arranged like the spokes of a wheel. There are also eight resistors in the wind vane as shown in Figure 3.9. As the magnet rotates, different reed switches will open and close and thus switch their corresponding resistor in and out of the circuit. Resistors can have different resistance values, reported in ohms Ω and those with low resistance let almost all current through, while those with high resistance let very little current through. The most common uses for resistors are protecting components from being damaged by currents that are too high, or dividing voltage between different parts of a circuit. Each of the eight resistors has a different value, and this allows the wind vane to have 16 possible combinations of resistance, since the magnet can close two neigboring reed switches when it's positioned halfway between them. In order to read the wind direction from the vane, instead of measuring the resistance value directly, it is actually much easier to record a voltage from the wind vane that varies according to which combination of resistors is currently switched into the circuit. This means by measuring an analogue value, the wind vane will continuously report a range of voltages. By contrast, the anemometer simply reports a 'HIGH' or 'LOW' voltage, meaning it sends a digital signal by converting it from analog signal from rotation of wind vane.

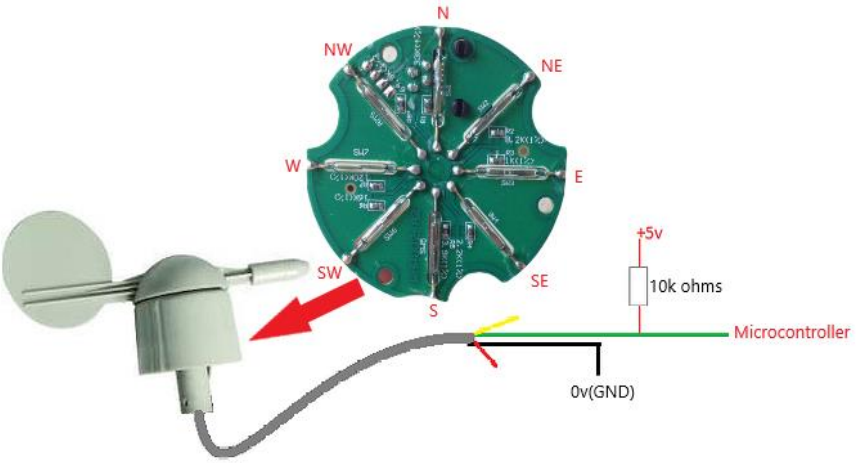


Figure 3.9 Wind Vane internal component

3.3.8 Rain Gauge

Rain gauges are self-emptying tipping bucket-type mechanical devices that measure precipitation in millimetres of height gathered on one square metre over a specific time period. Rain is gathered and directed into a bucket. When the bucket has collected enough rainwater, it will tip over, the water will drain from the bottom, and the opposite bucket will rise into position. Each 0.011" (0.2794 mm) of rain results in a single instantaneous contact closure, which can be recorded using a digital counter or microcontroller interrupt input. To compute the amount of rainfall, multiply this precise value by the number of tips. The switch on the gauge is linked to the two centre conductor wires of the associated RJ11-terminated cable. Rain collected across a 54-square-centimetre area pours into the bucket, which flips over when full. A small cylindrical magnet points towards the back wall inside the ridge between the two buckets. As the bucket tips, a magnet in the apex causes a glass-encapsulated reed-relay in a housing behind the bucket to close, as shown in Figure 3.10. The rain gauge, like the anemometer, can be treated as a button by attaching it to a GPIO pin on the microcontroller and counting the number of 'presses' to compute rainfall.

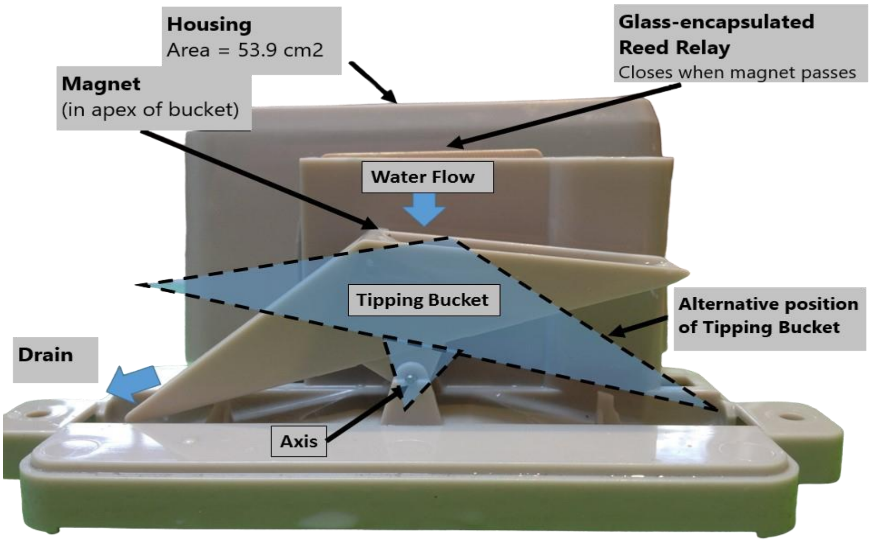


Figure 3.10 Rain Gauge internal mechanism

3.3.9 DHT21 Temperature & Humidity Sensor

DHT21 is a digital temperature and humidity module that contains a compound with a calibrated digital signal output from a temperature and humidity sensor. The use of a dedicated digital module collection technology as well as temperature and humidity sensor technology ensures the product's high dependability and long-term stability. The sensor is made up of capacitive sensor wet components and high-precision temperature measuring devices that are linked to a powerful 32-bit microprocessor. The device is of great quality, has a quick response time, a good anti-jamming capabilities, and is expensive. Each sensor is highly accurate in terms of humidity calibration chamber calibration. The calibration coefficients are called in the form of procedures, the calibration coefficients stored in the microcontroller, the sensor inside the processing of the heartbeat. System integration is quick and easy thanks to the standard single-bus interface. Small size, low power consumption, and signal transmission distances of up to 20 metres make it the perfect choice for all types of applications, including the most demanding. It has only three wires, including power and ground, as shown in Figure 3.11, implying that only one digital pin is necessary to interface it to a microcontroller. The DHT library, which is available through the Arduino IDE, provides excellent support for the sensor on Arduino development platforms.

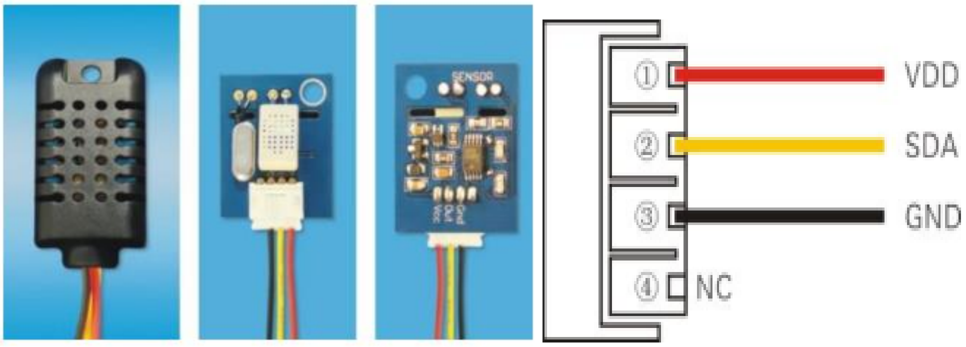
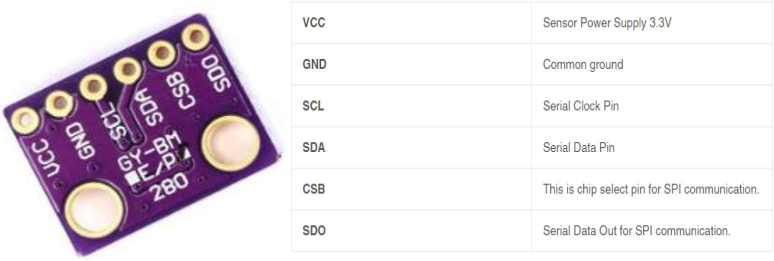


Figure 3.11 DHT21

3.3.10 BMP280 Barometric Pressure

The BMP280 absolute barometric pressure sensor is a tiny device ideal for a wide variety of uses. The module for sensing is contained in a very small metal enclosure. Because of its compact size and low power requirements, it can be used in a battery-operated configuration. Bosch's Piezo-resistive pressure sensor technology forms the basis of the BMP280, which boasts high precision, linearity, long term stability, and EMC resilience. Power consumption, resolution, and filter performance may all be fine-tuned with the device's numerous operating choices. The developer is given a tried-and-trued default set of parameters for a sample use case to facilitate design-inThis sensor able to connected to a Wi-Fi module making it compatible to collect data by sending it to cloud for home weather station or weather forecast application. The specification of BMP280 pinout is shown in Figure 3.12



VCC
GND
SCL
SDA
CSB
SDO
GY-BME/P
280

VCC	Sensor Power Supply 3.3V
GND	Common ground
SCL	Serial Clock Pin
SDA	Serial Data Pin
CSB	This is chip select pin for SPI communication.
SDO	Serial Data Out for SPI communication.

Figure 3.12 BMP280 Pinout

3.4 Milestone 2: Software Development

In this section will explain about the software that to be used to combine with the hardware module. The most important is the coding and the simulation to do the project as it will observe overall system working. Flowchart for this milestone 2 is shown in Figure 3.13.

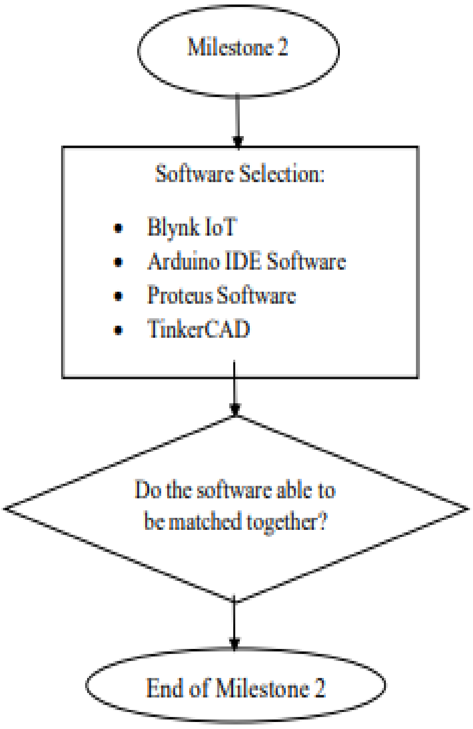


Figure 3.13 Flowchart for Milestone 2

3.4.1 Blynk IoT

Blynk IoT was created mainly for the Internet of Things (IoT). It can manage hardware remotely, show sensor data, store data, display it, and perform a variety of other fascinating things over iOS or Android smartphones with internet connectivity. With a user-friendly interface that capable of displaying the data graphically, and a convenient solution for configuration, code creation, uploading and visualization. To use the Blynk IoT, a cloud compatible board is required which is ESP32 microcontroller-based board. Three major components in the platform are Blynk App which allows to create amazing interfaces for projects using various widgets that has been provided. Secondly, Blynk Server that is responsible for all the communications between the smartphone and hardware. Blynk Cloud works by run private Blynk server locally. Thirdly, Blynk Libraries allow for all the recognize hardware platforms to enable communication with the server and process all the incoming and outcoming commands. Figure 3.14 shows working flow of Blynk app.

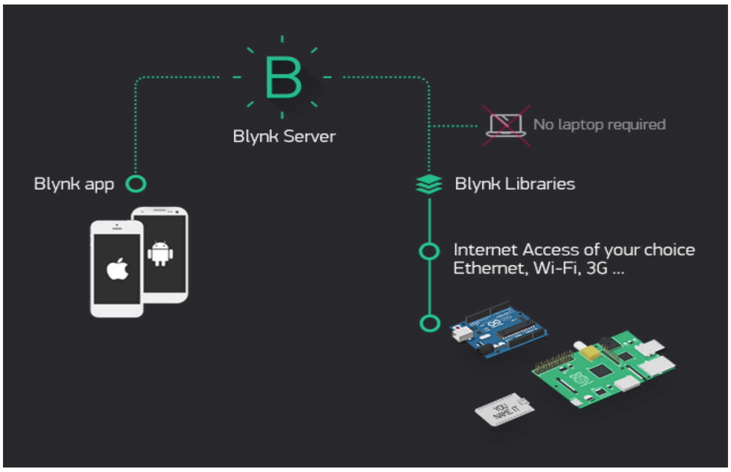
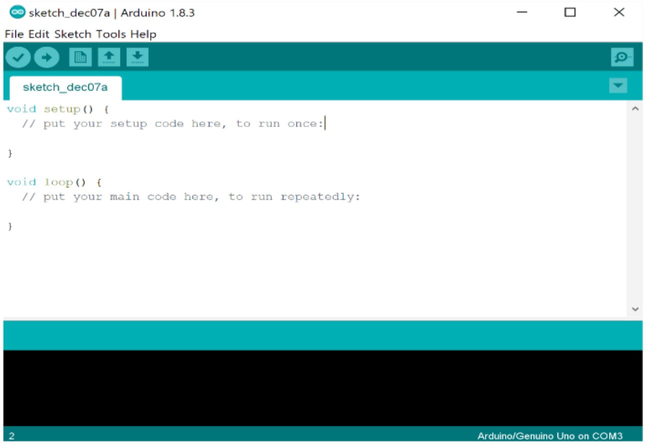


Figure 3.14 Blynk IoT

3.4.2 Arduino IDE Software

The coding will be written in the Arduino programming, which is known as C++ language. It is easy to use for beginners due to it simplifies process of working with microcontrollers. Arduino IDE software will be used as it is a special software running that allows writing the Arduino language. Arduino IDE Software used to compile the programming coding and convert it into binary form and transfer it to the circuit board by using serial port communication as shown in Figure 3.15. With the Arduino that used C++ programming language which is very simple and makes coding studying easier as it used standard serial protocol communication, so it can be directly connects tocomputer via USB to compile the program to hardware. In addition, the Arduino softwareallows users to connect pre-built circuit board, which called as shields that provide additional capabilities that enable users to explore more variations sensors, displays, and inputs.


sketch_dec07a | Arduino 1.8.3
File Edit Sketch Tools Help
sketch_dec07a
2
Arduino/Genuino Uno on COM3

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure 3.15 Arduino IDE Software Interface

3.4.3 Proteus Software

To designing the electrical circuit for his project Proteus software has been chosen as its easier to accumulate. Proteus software as shown in Figure 3.16 is used to simulate, design and drawing of electronic circuits. Proteus mainly used by the design engineers and technician to created schematicdrawing and electronic printed circuit board. In this software simulation, the possibility of circuit faulty is lower such as a loose connection that takes a long time to discover connection problems in a real circuit. So, using this software is pretty convenient to test and design project circuit.

Figure 3.16 Proteus Software

3.4.4 TinkerCAD

The basic prototype for this based home weather station equipment is designed using 3D TinkerCAD shown as Figure 3.17. Wind vane is constructed as wind direction measurement and 3 cup anemometer as wind speed measurement. Rain gauge as rainfall measurement detection. There also enclosure box which is designed to place all the electronic components' part inside. All this design is just a housing and as mechanism part. Electrical sensory part will be place inside the housing of this mechanical part to measure weather parameter.

Figure 3.17 TinkerCAD

3.5 Milestone 3: Project Prototype Development

The milestone 3 is created by referring to third objective, which is developing the prototype of housing, wind vane, 3 cup anemometer and PCB board layout that can fit all together. The flowchart for milestone 3 is shown in Figure 3.18.

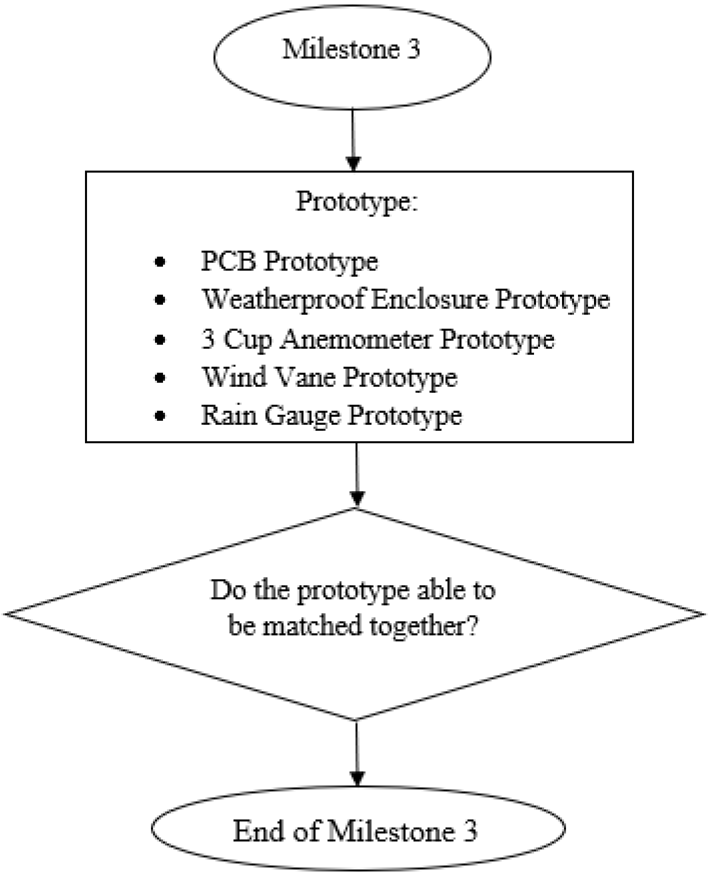

Milestone 3
Prototype:
PCB Prototype
Weatherproof Enclosure Prototype
3 Cup Anemometer Prototype
Wind Vane Prototype
Rain Gauge Prototype
Do the prototype able to be matched together?
End of Milestone 3

Figure 3.18 Flowchart for Milestone 3

3.5.1 PCB Prototype

After the system components configuration for this weather station system is well constructed and assigned, then the coding is programmed into the system to check the system function is running without error. To locate those system components in orderly, PCB boards with wiring are designed to fit all components and a housing is created to avoid any loose contact that might happen using breadboard. The PCB boards is designed by using Proteus Design Suite software, which is a computerized software tool that mainly used for electronic components design. First, a schematic diagram for this weather station system is drawn inside the Schematic Capture and all the pinout of the chip is referred to respective datasheet. After that, the process of arrangement of the components into a desired dimension PCB board is required in PCB Layout tab. Then followed the trace line that connected between components is link as shown in figure 3.19. Lastly, the virtual PCB board is created and able to view in Proteus software for preliminary result.

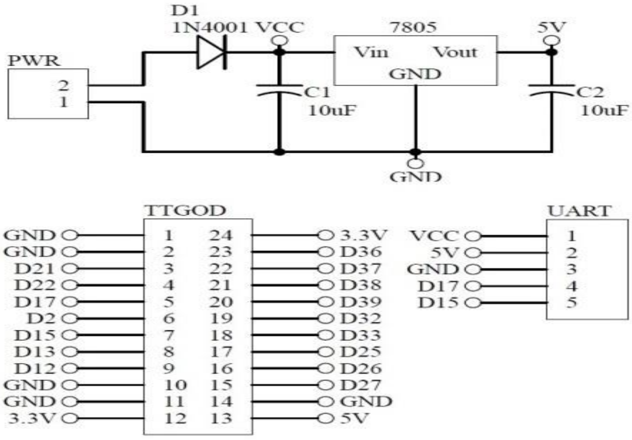


Figure 3.19 components connection link

After succesfully designed PCB board based on all criteria, the board is ordered online by sending the design to seller to be printed at China according to seller. The PCB board undergoes a unique PCB board development process there by cut PCB board into size, etching process, apply PCB mask, UV light cure PCB mask and finally Drill PCB board to finalize the process. Figure 3.20 shows a final design of PCB board that has been designed.

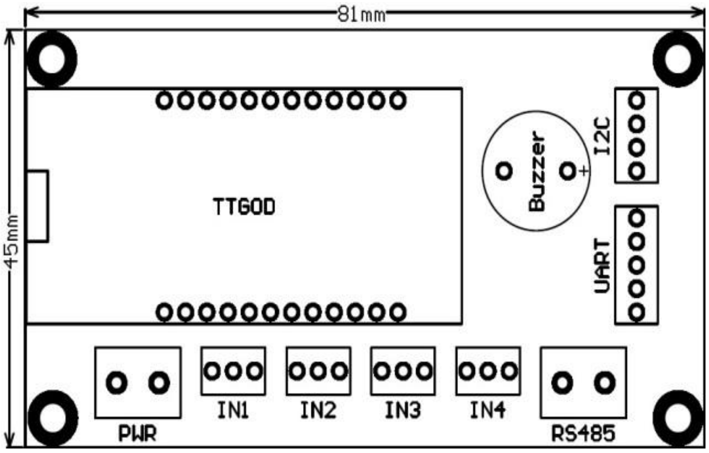


Figure 3.20 PCB board design

3.5.2 Weatherproof Enclosure Prototype

Weatherproof enclosure is to house the external environmental sensors from gets wet or even very damp. This case can hold and fit PCB board or electronic components inside by become enclosure for wire junction. The prototype can act as housing that use plastic as main components because it is convenient in price and lightweight. For the design of this plastic housing, TinkerCAD software is used for drawing the plastic housing. The key idea is to allow outside air to flow around the sensors but to prevent moisture from reaching them. This software is a solid modelling computer aided design. While using this software to create the drawing, it gives early sight on how it can allows users to place weather station easily and if any circumstances occur, users can remove it back with ease. A sample of using the TinkerCAD software to draw the overall face of the plastic housing is shown at Figure 3.21. Its also need to provide waterproofing characteristics and offer a simple low-cost solution for general installation use. This box need to be waterproof enough to be at outdoor to assure any electronic components and sensitive electrical connections that placed inside protected and secure in any environment.

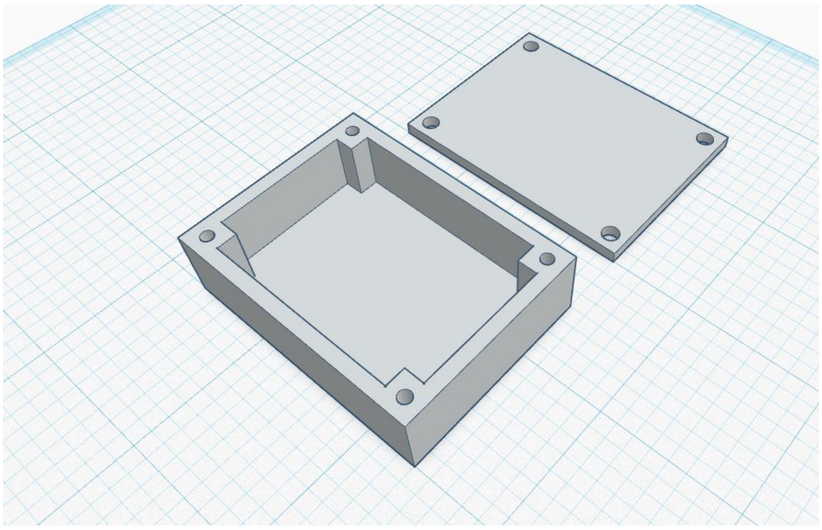


Figure 3.21 Components Enclosure

3.5.3 3 Cup Anemometer Prototype

For the design of this 3 cup anemometer, TinkerCAD software is used for drawing the three arms with scoops on the end. The key idea is to allow these 3 cup to catch the wind and cause the arms to spin. This software is a solid modelling computer aided design. While using this software to create the drawing, it allows to give early insight on how the mechanism of this 3 cup anemometer working. It can be use as active mechanical devices that physically interact with the environment to measure accurate real-time wind speed. A space for reed switches and a rotating magnet to be place inside the anemometer is reserved. A sample of using the TinkerCAD software to draw the overall side of the 3 cup anemometer is shown at Figure 3.22.

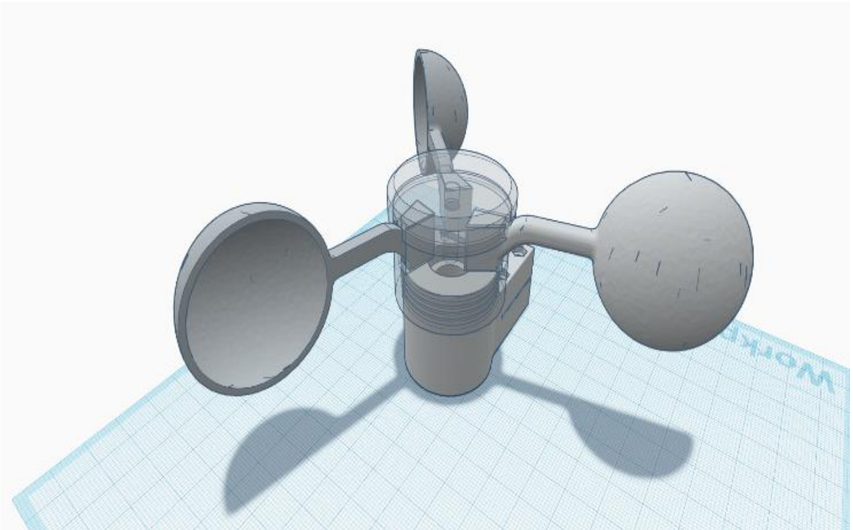


Figure 3.22 3 Cup Anemometer prototype

3.5.4 Wind Vane Prototype

For the design of this wind vane, TinkerCAD software is used for drawing the pointing direction of the wind coming from and circular back to balance the working mechanism. The key idea is to allow wind vane works when wind exerts force on its vertical blade, which rotates to find the position of least wind resistance thus making this position is aligned with the direction of the oncoming wind. This software is a solid modelling computer aided design. While using this software to create the drawing, it allows to give early insight on how the mechanism of this wind vane working. It can be act as active mechanical devices that physically interact with the environment to measure accurate real-time wind direction. A space for reed switches and a rotating magnet to be place inside the wind vane is reserved. A sample of using the TinkerCAD software to draw the overall side of the wind vane is shown at Figure 3.23.

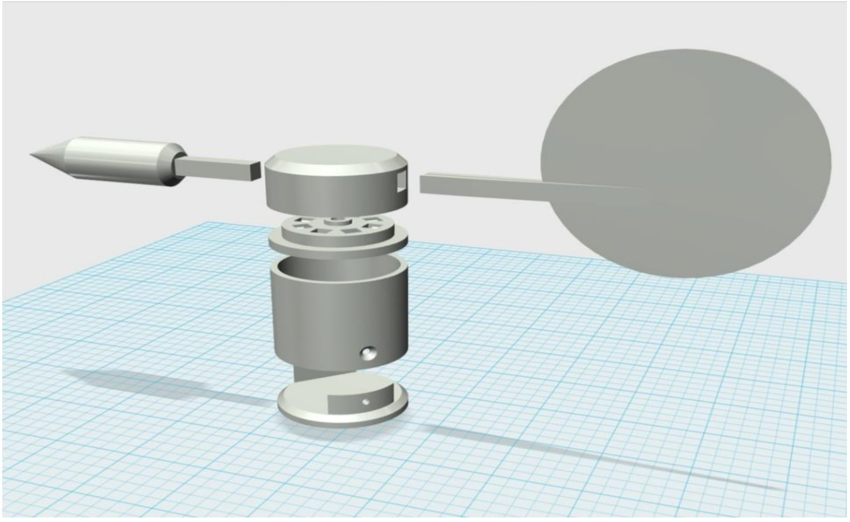


Figure 3.23 Wind Vane prototype

3.5.5 Rain Gauge Prototype

For the design of this rain gauge, TinkerCAD software is used for drawing self-emptying tipping bucket type that measure precipitation. The key idea is to allow rain to be captured over an area of about 54 square centimetres falls into the bucket which tips over when it is full. This software is a solid modelling computer aided design. While using this software to create the drawing, it allows to give early insight on how the working mechanism of rain gauge. The rain gauge uses a tipping bucket mechanism which its can act as active mechanical devices that physically interact with the environment to measure accurate amount of rainfall. A space for magnet in the apex of the bucket and glass-encapsulated reed-relay to be place at the back of the rain gauge is reserved. A sample of using the TinkerCAD software to draw the overall side of the rain gauge is shown at Figure 3.24.

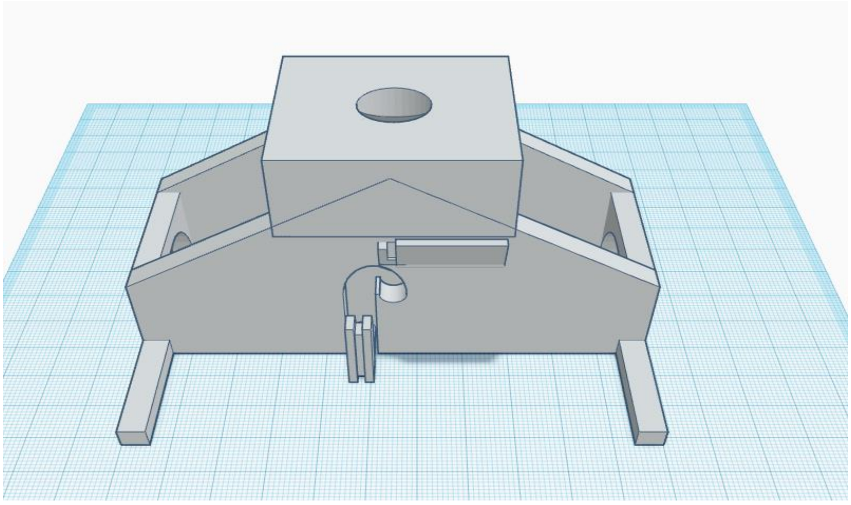


Figure 3.24 Rain Gauge prototype

3.6 Milestone 4: System Performance Analysis

The milestone 4 is created by referring to forth objective, which is analyzing the performance of the system by collecting the data with different method. Then the data collected will be analyzed to unsure the performance of the system suitable to measure weather parameter. If there is weak in performance, modification will made to the system by return to previous milestones. The flowchart for milestone 4 is shown in Figure 3.25.

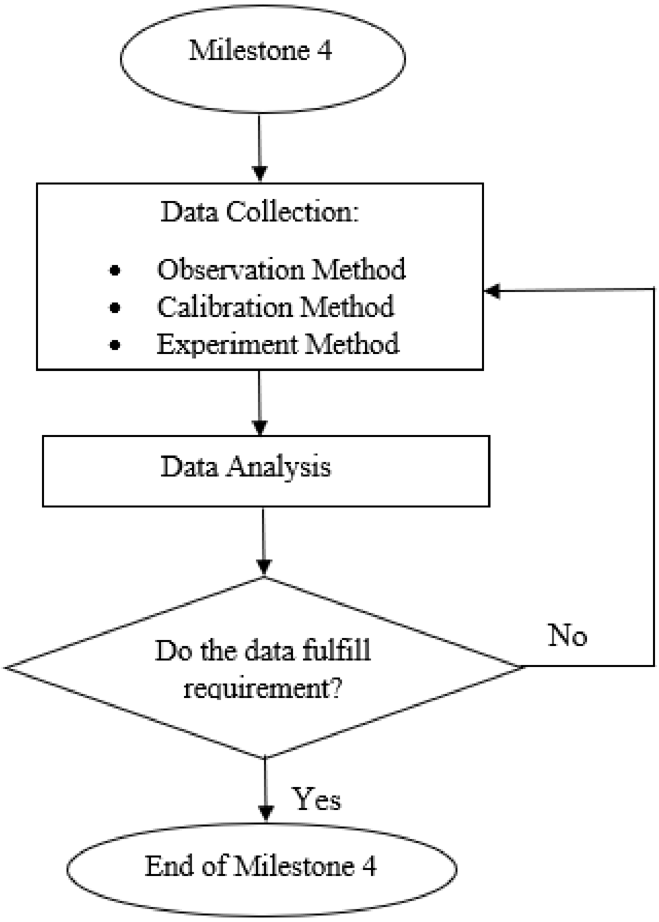


Figure 3.25 Flowchart for Milestone 4

3.6.1 Data Collection

Experimental collection is a method of gathering and analyzing data from many sources in order to produce a complete and accurate explanation that will ensure the quality of this weather station equipment. There are several method used to collect the data required for this project.

3.6.1.1 Observation Method

This observation is the most common and simplest method since it does not required much technical support. First, datasheet of the components used in this system is revised to gather those important information to achieve the optimum results from the system. After the prototype created, observation is made on it by understanding all the function either good or bad to use.

3.6.1.2 Calibration Method

Calibration is one of the primary processes used to maintain instrument accuracy. Calibration is the process of configuring an instrument to provide a result for a sample within an acceptable range. Eliminating or minimizing factors that cause inaccurate measurements is a fundamental aspect of instrumentation design. In this project, calibration for each sensory part is an important factor to determine full functionality of system. Although the exact procedure may vary from sensor to sensor, the calibration process generally involves using the instrument to test samples of one or more known values called “calibrators” . For example, to determine wind direction is by using digital compass while for wind speed is using digital anemometer to anlyse its accuracy. The results are used to establish a relationship between the measurement technique used by the instrument and the known values. The instrument can then provide more accurate results when samples of unknown values are tested in the normal usage of the product.

3.6.1.3 Experiment Method

This project required a complete circuit diagram then followed by Wi-Fi setup process. By using experimental method, different components or just changing the value will causes different output signal and this data is collected for comparison or modification on this system to make it become perfect.

3.6.5 Data Analysis

In this project, there are several data obtained during testing the system, such as the relationship between wind speed and direction, rainfall, temeprature, humidity and barometric pressure against reliable real-time weather website. The observed data is recorded through Blynk app and the graph is plotted easily and quickly inside it. Then, the graph of data is analyzed to draw helpful conclusion from the data obtained.

3.7 Summary

The methodology is formulated into four milestones in order to achieve the objectives of this project that design a high performance and low cost self-alignment tool. Milestone 1 was design the system architecture with the overall components configuration. Milestone 2 developed the software choice for the system function well without error. These milestone will be explained more detail in the result and analysis part. After that, the milestone 3 created the prototype of the system to make it solid. In order to achieve the best performance, data collection and analysis conducted in last milestone, which is milestone 4. This chapter ensure all the progress undergoes on the flow orderly.

CHAPTER 4

RESULTS & ANALYSIS

4.1 Introduction

This section describes the results, analysis, and discussion for all the data collected from the system to determine the performance of the system and modified it to achieve the best results for this weather station. In this chapter, it shows the result of hardware, calibration of wind parameters, durability of hardware prototype, weather station results with forecast website and the result analysis of weather station.

4.2 Result of Hardware and Software

The electric circuit was designed by using Proteus software. The result was split into two parts, which are virtual hardware design and physical hardware design. This system come with 3 cup anemometer for wind speed measurement, wind vane for wind direction measurement, rain gauge for rainfall measurement, DHT21 for temperature and humidity, and BME280 for barometric pressure measurement. ESP32 as device to pairing device for Wi-Fi capability and 32-bit microcontroller to record, store, measure and calculate data from sensor to be send into signal.

4.2.1 Virtual Hardware Design

The electric circuit was designed by using Proteus software as shown in Figure 4.1. The design was mainly having two parts, which is ESP32 as Wi-Fi connectivity for system and a 32-bit microcontroller to record sensors data. This system come with 3 Cup Anemometer and wind vane respectively to determine wind direction and Wind direction. There also rain gauge to measure rainfall per hour. Another sensory device such DHT21 and BMP280 as weather parameters such temperature, humidity, and barometric pressure. As noted, this system will be supply by power source of 5V. Inside system there will be voltage regulator to control any abnormal voltage inside the system. ESP32 have a port to plug USB-C to power up the device. The purpose of using 32-bit microcontroller is because of its arithmetic logic units, registers, and bus width. In general, this means that a 32-bit can handle quadruple the amount of data, making it technically more data efficient. This is perfect choice to store or handle such data for this weather station.

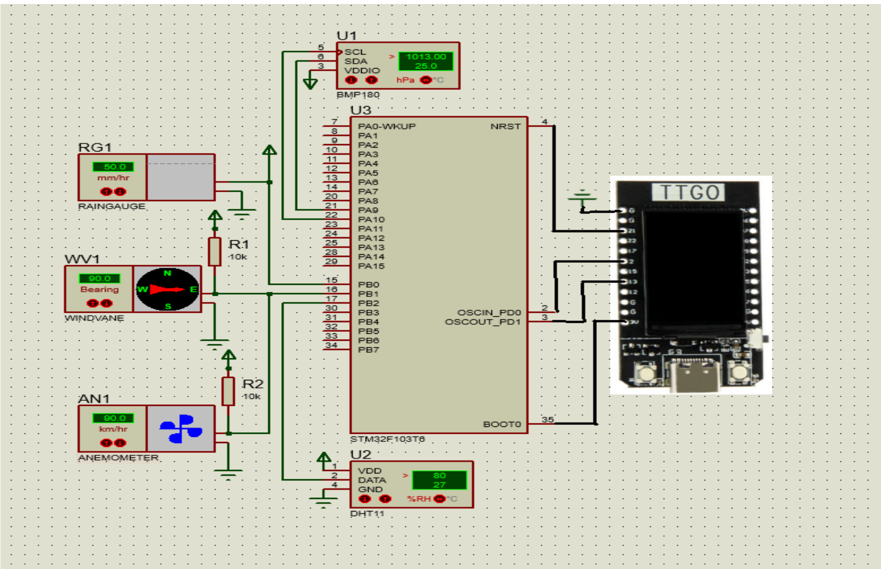


Figure 4.1 Circuit design in Proteus Software

4.2.2 Physical Hardware Design

As shown in Figure 4.2, this weather station contains several important parts in its design. PCB board that connected to all of sensor's device like wind vane, 3 cup anemometer, rain gauge, DHT21 and BMP280 is placed inside weatherproof enclosure. These external environmental sensors are ensured to be watertight to able withstand such blustery weather. The key idea is to allow outside air to flow around the sensors but to prevent moisture from reaching them. Underneath enclosure box in the middle of weather station have a couple of holes for the RJ11 cables connecting the wind and rain sensors, and one other hole for 5V power source USB-C wire. This enclosure with holes facing downward from the weather station can keep direct rainfall out. To get representative readings for ambient temperature and humidity, air needs to circulate around the DHT21 sensor. This is some challenging for this design of weather station as the main priority is to keep water from entering the enclosure so DHT21 must be place inside the enclosure although some inaccurate measurement might occur especially during a hot weather. By installing weather station outside, it can be done with mount installation on a wall, rooftop, fence, or even on a plumbing pipe stuck in the ground. Any location is fine as long as the sensors are exposed to the elements of weather parameter. It is not possible to provide specific instructions for mounting weather station, as the exact method will depend on particular location and environment to install it. The rain gauge needs to collect rain, 3 cup anemometer and wind vane need to be spinning to detect present of wind, DHT21 temperature and humidity sensor needs to get some fresh open air as well. This is quite challenging for this weather station design main objective which is to prevent any leakage from entering electronic components that placed inside enclosure. The solution is to drill some holes underneath weatherproof enclosure to avoid situating it in direct of rainfall. Besides, the weather station needs to be connected to 5V USB-C to power up the weather station. By this, weather station will instantly be operating and able to detect local network to send weather data for Blynk app via Wi-Fi connectivity.

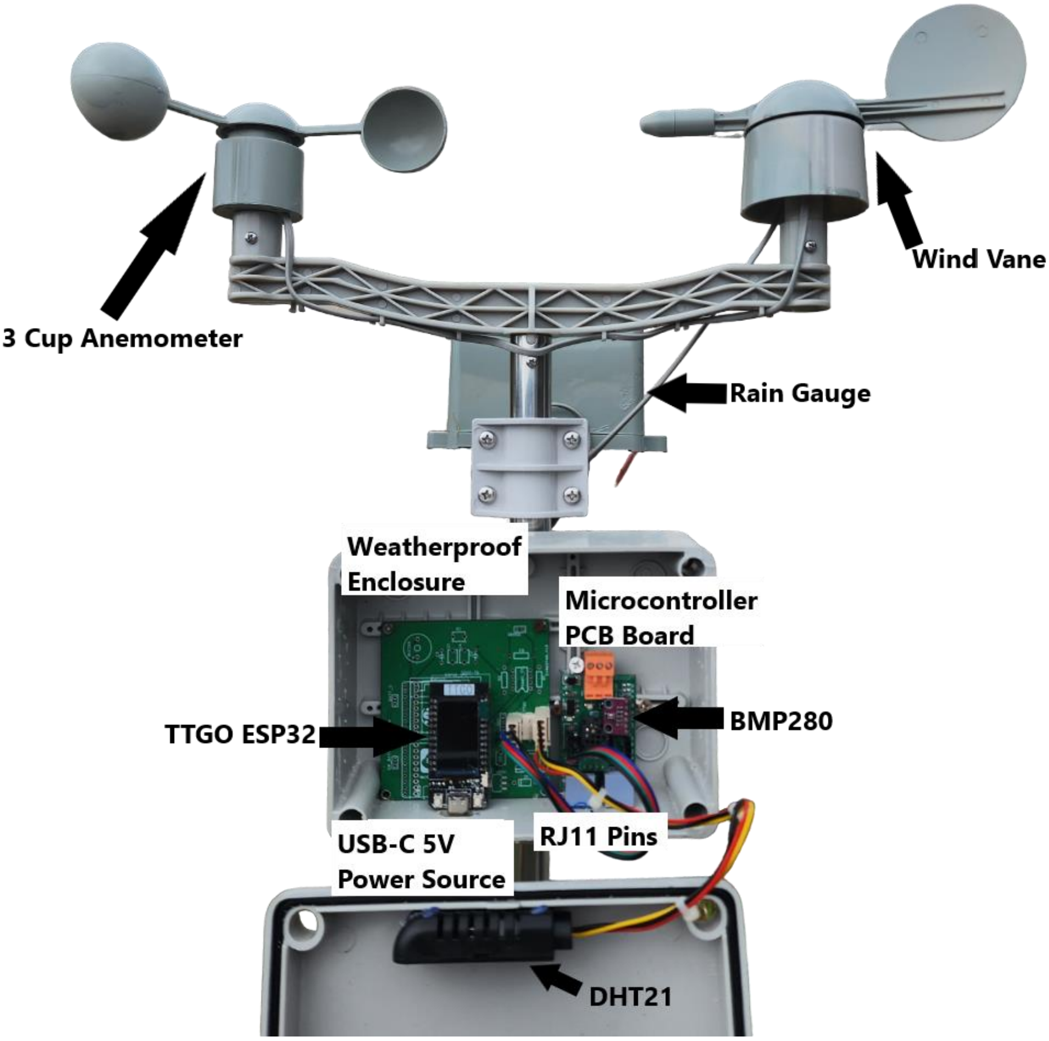


Figure 4.2 Finallize project design

4.2.4 Result of Wi-Fi connectivity to Blynk app

To make this weather station able to upload data to somewhere that can be view and analyze, it will need some form of internet connection. By using Wi-Fi which is a typically the easiest way to do this. Upon entering specific component details such as ESP32 IP address, it can be directly pairing to setup a device with Wi-Fi capability. By connecting to local Wi-Fi server that has been configure inside coding of Arduino IDE software, weather station will be able to have internet connectivity. However, the range of ESP32 need to be near to source of this Wi-Fi range. ESP32 reads sensor data from weather station sensor and sends it to the Blynk server. Wireless access can be used in conjunction with a weather station if the Wi-Fi range is optimal. Figure shown the successful of Wi-Fi connectivity from ESP32 to Blynk app. As shown in Figure 4.3, Blynk app resulting in turning online if there is connectivity between ESP32.

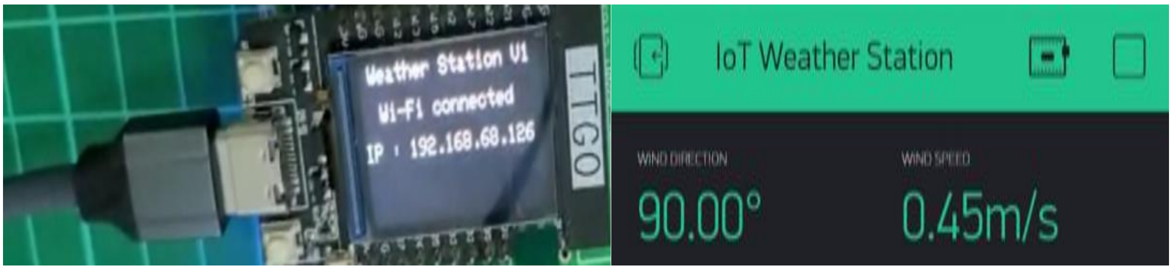
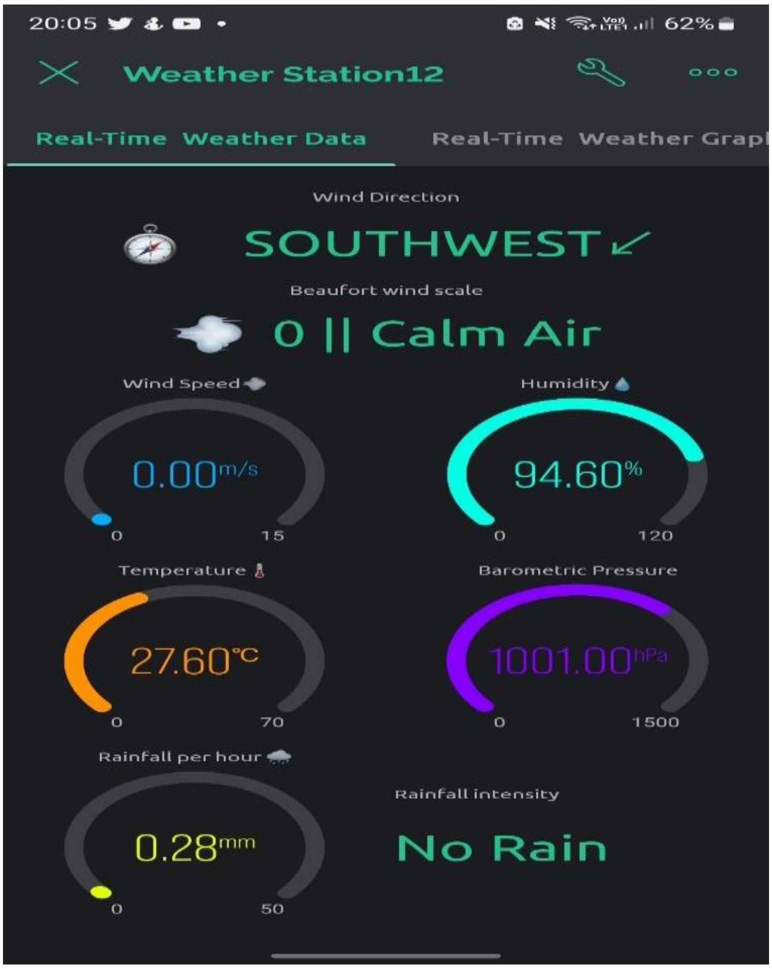


Figure 4.3 Successfully link ESP32 to Blynk app

4.2.5 Blynk IoT app interface design

The design of Blynk app interface is to create a graphical interface or human machine interface (HMI) by compiling and providing the appropriate address on the available widgets. Blynk was designed for the Internet of Things which can monitor hardware remotely, it can display sensor data, it can store data, visualize it. Thus, making the Blynk app as a user-friendly IoT software and can be freely design interface in it such as graph, timetable, and even real-time data. A database is a good place to keep weather records. Keeping weather station data in a database allows for efficient storage of massive amounts of information and facilitates easy access, retrieval, and analysis of that information. Figure 4.4 shows weather station Blynk app interface.



Wind Direction

SOUTHWEST ↙

Beaufort wind scale

0 || Calm Air

Wind Speed 0.00m/s

Humidity 94.60%

Temperature 27.60℃

Barometric Pressure 1001.00hPa

Rainfall per hour 0.28mm

Rainfall intensity

No Rain

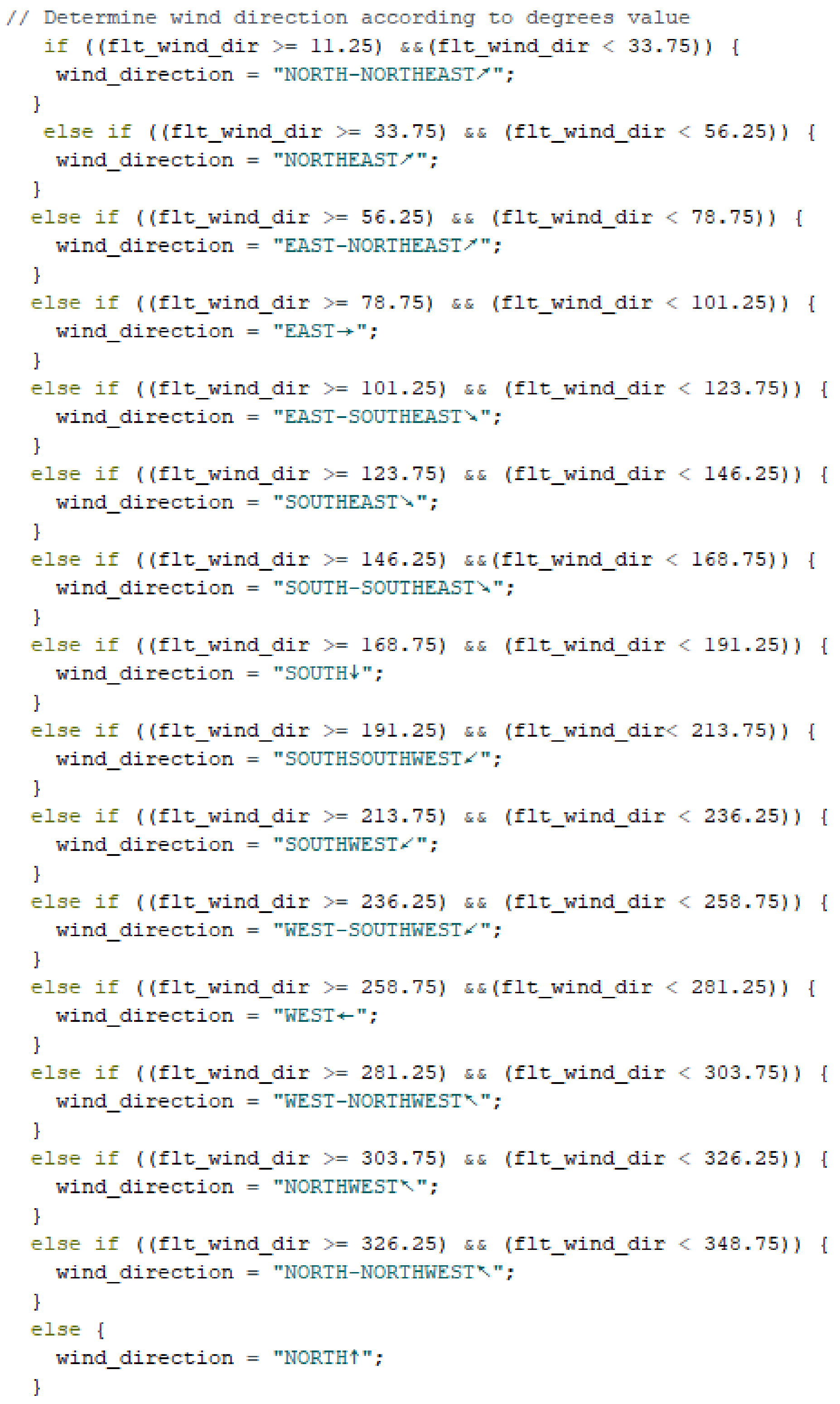
Figure 4.4 Blynk app interface design

4.3 Calibration Result of Weather Station

The calibration of the wind direction and wind speed result is verified with specific method such using precise digital anemometer for wind speed and digital compass for wind direction. Within these methods, this weather station can be proved as accurate to measure weather parameter so it can be test outside to analyze its data. And as for durability of weather station against such weather can be tested by try to let the weather station operating outside for some period of time. The weather station calibration is separate into three parts, which are calibration of wind speed, wind direction and durability of weather station when install it outdoor.

4.3.1 Wind Direction Coding Function

Before calibrating the wind direction, the overall program for wind direction is sketched inside the Arduino IDE software. A lot of troubleshoot has been done to ensure the system working accordingly. The main important thing is to setup in this coding line is to initialize every related possible sensor library into Arduino IDE software. Then, proceed with declaring it into the very first line of coding. After register auth token, ssid and its pass for Blynk part, the other coding flow can be done seamlessly. One of the challenging parts in coding for this weather station is to determine the angle degrees for wind direction measurement. In every full rotation degree of 360°, the direction of wind needs to be assigned respectively. For every each of these degrees value is then converted into string value in the coding to able it to display as string character on Blynk app. First step is declaring its float and string value. There are 8 positions of wind direction to be indicated for this system so the range for every direction will be 45°. When the degree that has been recorded is vary from 11.25° to 56.25° indicated based on point of wind vane, the wind direction will be register as Northeast. Wind direction is register as East when the degree is from 56.25° to 101.25°. Wind direction is register as East when the degree is from 56.25° to 101.25°. register as Southeast when the degree is from 101.25° to 146.25°. Wind direction is register as South when the degree is from 146.25° to 191.25. Wind direction is register as Southwest when the degree is from 191.25° to 236.25°. Wind direction is register as West when the degree is from 236.25° to 281.25°. Wind direction is register as Northwest when the degree is from 281.25° to 326.25°. And lastly, the remaining degrees value is register as North. Figure 4.5 shows coding function for wind direction.



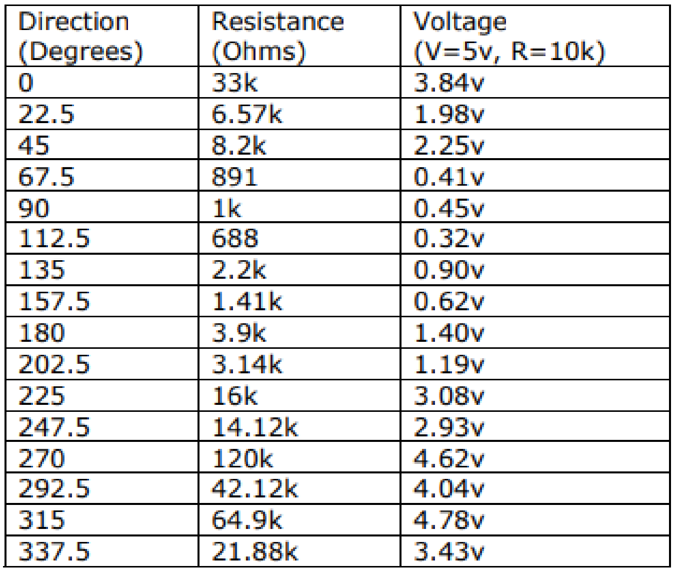
```
// Determine wind direction according to degrees value
   if ((flt_wind_dir >= 11.25) &&(flt_wind_dir < 33.75)) {
    wind_direction = "NORTH-NORTHEAST↗";
  }
   else if ((flt_wind_dir >= 33.75) && (flt_wind_dir < 56.25)) {
    wind_direction = "NORTHEAST↗";
  }
  else if ((flt_wind_dir >= 56.25) && (flt_wind_dir < 78.75)) {
    wind_direction = "EAST-NORTHEAST↗";
  }
  else if ((flt_wind_dir >= 78.75) && (flt_wind_dir < 101.25)) {
    wind_direction = "EAST→";
  }
  else if ((flt_wind_dir >= 101.25) && (flt_wind_dir < 123.75)) {
    wind_direction = "EAST-SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 123.75) && (flt_wind_dir < 146.25)) {
    wind_direction = "SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 146.25) &&(flt_wind_dir < 168.75)) {
    wind_direction = "SOUTH-SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 168.75) && (flt_wind_dir < 191.25)) {
    wind_direction = "SOUTH↓";
  }
  else if ((flt_wind_dir >= 191.25) && (flt_wind_dir< 213.75)) {
    wind_direction = "SOUTHSOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 213.75) && (flt_wind_dir < 236.25)) {
    wind_direction = "SOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 236.25) && (flt_wind_dir < 258.75)) {
    wind_direction = "WEST-SOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 258.75) &&(flt_wind_dir < 281.25)) {
    wind_direction = "WEST←";
  }
  else if ((flt_wind_dir >= 281.25) && (flt_wind_dir < 303.75)) {
    wind_direction = "WEST-NORTHWEST↖";
  }
  else if ((flt_wind_dir >= 303.75) && (flt_wind_dir < 326.25)) {
    wind_direction = "NORTHWEST↖";
  }
  else if ((flt_wind_dir >= 326.25) && (flt_wind_dir < 348.75)) {
    wind_direction = "NORTH-NORTHWEST↖";
  }
  else {
    wind_direction = "NORTH↑";
  }
```

Figure 4.5 Wind Direction Coding

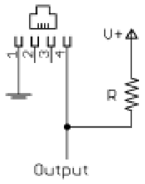
4.3.2 The Calibration of Wind Direction

Calibration of wind direction is based on ADC (Analog to Digital converter) to represent direction of the wind coming from. It has eight switches, each connected to a different resistor. The vane's magnet may close two switches at once, allowing up to 16 different positions to be indicated. An external resistor can be used to form a voltage divider, producing a voltage output that can be measured with an analog to digital converter, as shown below. The switch and resistor arrangement is shown in the diagram to the right. Resistance values for all 16 possible positions are given in the table 4.1. Resistance values for positions between those shown in the diagram are the result of two adjacent resistors connected in parallel when the vane's magnet activates two switches simultaneously.

Table 4.1: Conversion each resistance value to Direction (Degrees)



Direction (Degrees)	Resistance (Ohms)	Voltage (V=5v, R=10k)
0	33k	3.84v
22.5	6.57k	1.98v
45	8.2k	2.25v
67.5	891	0.41v
90	1k	0.45v
112.5	688	0.32v
135	2.2k	0.90v
157.5	1.41k	0.62v
180	3.9k	1.40v
202.5	3.14k	1.19v
225	16k	3.08v
247.5	14.12k	2.93v
270	120k	4.62v
292.5	42.12k	4.04v
315	64.9k	4.78v
337.5	21.88k	3.43v



Example wind vane interface circuit. Voltage readings for a 5 volt supply and a resistor value of 10k ohms are given in the table.

Based on wind vane interface circuit as shown in figure 4.6, it is noticeable that every direction is fully depend on resistance value in the range of each degrees respectively. Table 4.2 shown result of calibration wind direction of weather station based on digital compass angular value.

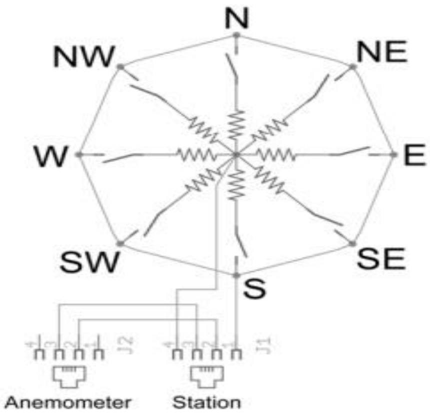
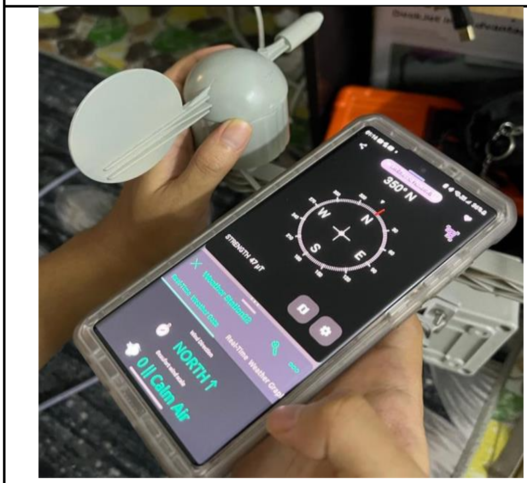
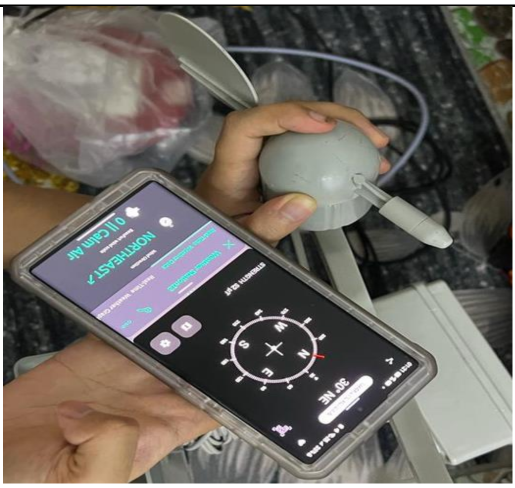
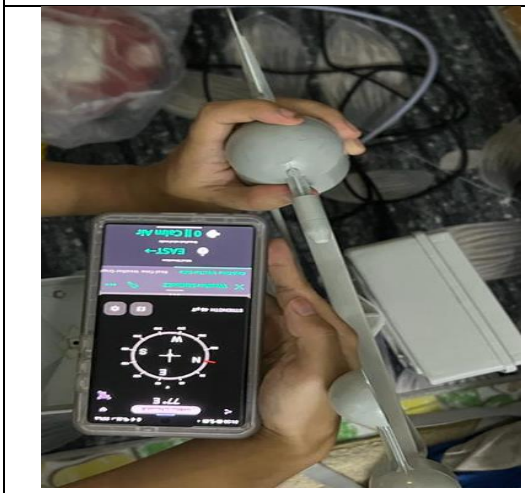
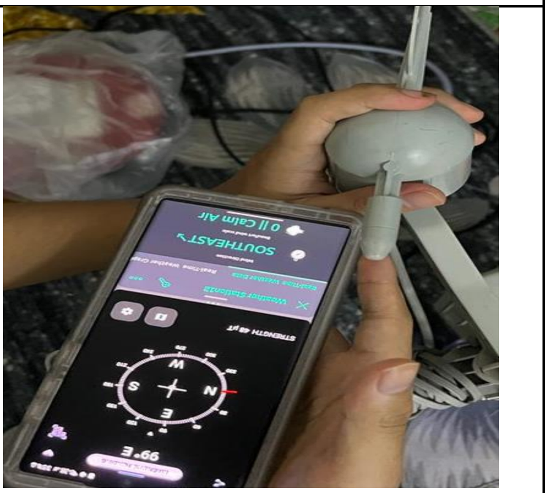
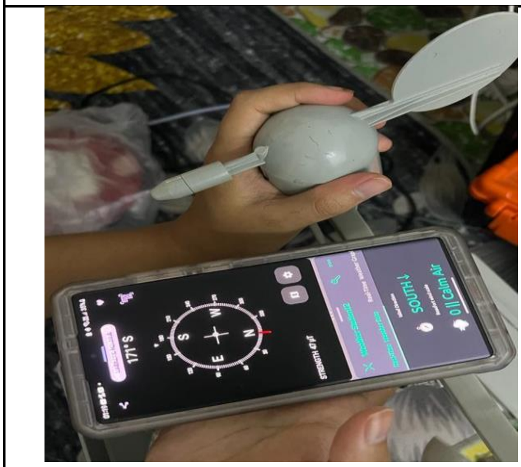


Figure 4.6 Wind Vane Interface Circuit

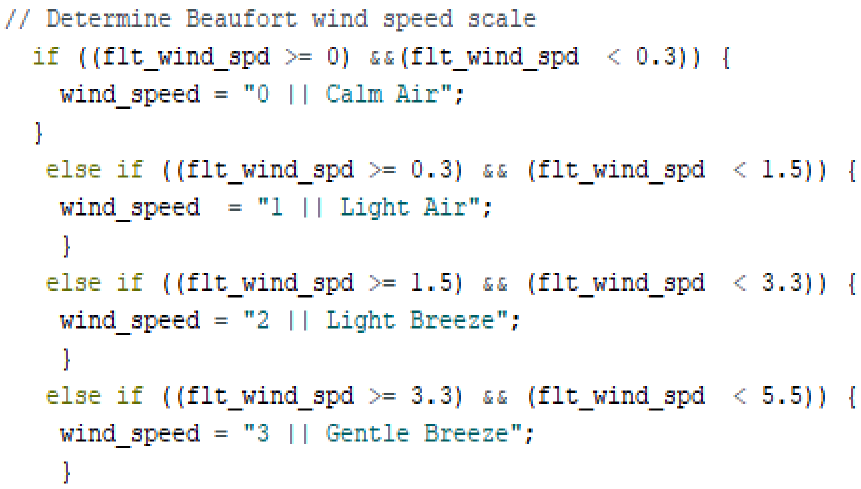
Table 4.2: Calibrated wind direction based on digital compass degree.

North = 350°	**NorthEast = 30°**
	
East = 77°	**SouthEast = 99°**
	

South = 171°	**SouthWest = 206°**
 171° S S W N E STRENGTH 42 µT SOUTH ↓ 0 || Calm Air	206° SW S W N E STRENGTH 45 µT WeatherStation02 SOUTHWEST↙ 0 || Calm Air
West = 254°	**NorthWest = 292°**
254° W S W N E STRENGTH 53 µT WeatherStation02 Real-Time Weather Data Real-Time Weather Graph Wind Direction WEST← Beaufort wind scale 0 || Calm Air	292° W S W N E STRENGTH 49 µT WeatherStation02 Real-Time Weather Data Real-Time Weather Graph Wind Direction NORTHWEST↖ Beaufort wind scale 0 || Calm Air

4.3.3 Wind Speed Coding Function

As for wind speed, it has been assigned to be record its value for every 1 minute. This to assure only consistent wind speed can be allowed to be measure in that period thus ensure only accurate wind speed is recorded. In this case, only wind speed of 1.492 MPH, 2.4 km/h or 0.67 m/s can cause the switch to close once per second. First step is declaring its float and string value. Then, referring to Beaufort wind scale to determine its force level according to reading of wind speed in m/s. As this weather station is marketed to use in household area, only until force level 3 is measured in this project. Inside the code, it will continually record the wind speed every 1 minute, keeping track of the largest measurement and calculating the average speed to be display as wind speed data. By setup wind speed float value accordingly to Beaufort scale, certain string value will be able to determine wind condition and its force level. Figure 4.7 shows coding function for this wind speed.



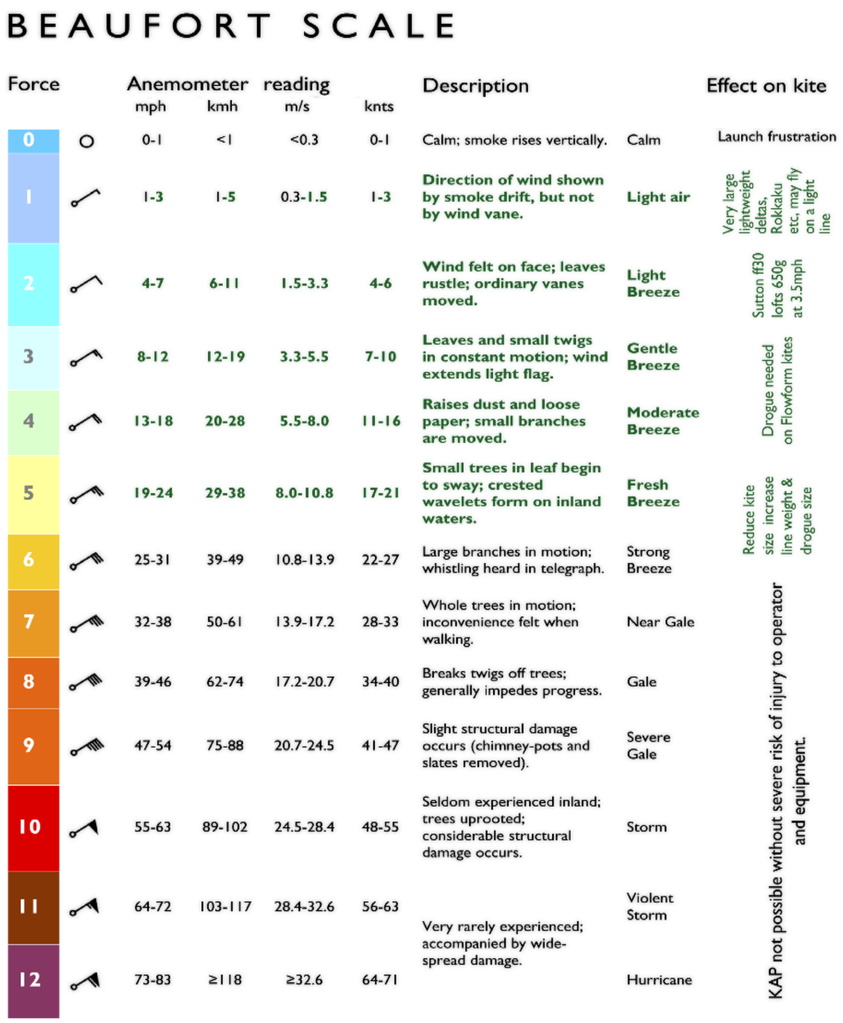
```
// Determine Beaufort wind speed scale
  if ((flt_wind_spd >= 0) &&(flt_wind_spd  < 0.3)) {
    wind_speed = "0 || Calm Air";
  }
   else if ((flt_wind_spd >= 0.3) && (flt_wind_spd  < 1.5)) {
    wind_speed  = "1 || Light Air";
    }
   else if ((flt_wind_spd >= 1.5) && (flt_wind_spd  < 3.3)) {
    wind_speed = "2 || Light Breeze";
    }
   else if ((flt_wind_spd >= 3.3) && (flt_wind_spd  < 5.5)) {
    wind_speed = "3 || Gentle Breeze";
    }
```

Figure 4.7 Wind Speed Coding

4.3.4 The Calibration of Wind Speed

The 3 cup-type anemometer measures wind speed by closing a contact as a magnet moves past a switch. A wind speed of 0.67 m/s causes the switch to close once per second. The anemometer switch is connected to the inner two conductors of the RJ11 cable shared by the anemometer and wind vane (pins 2 and 3.). It is needed to calibrate project weather station wind speed to verify its data. Wind speed is recorded in every 1 minute of rotation. Wind speed is then calculated based on speed of wind blowing the anemometer cup. Some error such loss of accuracy is due to something called the anemometer factor and is a result of some of the wind energy being lost when the arms turn. To compensate for this, wind speed can be multiply the reading generated by wind speed of 0.67 m/s that causes the switch to close once per second by an adjustment factor. Store the anemometer adjustment value as a constant with the value of wind speed 0.67 m/s in Arduino IDE coding and then use the constant in wind speed calculation. The anemometer produces two signals per spin, so can count the number of full rotations of the sensor by halving the number of detected inputs. This can then be used to calculate the wind speed such speed = distance of wind blowing / time.

Table 4.3: Beaufort scale



BEAUFORT SCALE

Force	Anemometer	reading			Description		Effect on kite
	mph	kmh	m/s	knts			
0	0-1	<1	<0.3	0-1	Calm; smoke rises vertically.	Calm	Launch frustration
1	1-3	1-5	0.3-1.5	1-3	Direction of wind shown by smoke drift, but not by wind vane.	Light air	Very large lightweight deltas, Rokkaku etc, may fly on a light line
2	4-7	6-11	1.5-3.3	4-6	Wind felt on face; leaves rustle; ordinary vanes moved.	Light Breeze	Sutton ff30 lofts 650g at 3.5mph
3	8-12	12-19	3.3-5.5	7-10	Leaves and small twigs in constant motion; wind extends light flag.	Gentle Breeze	Drogue needed on Flowform kites
4	13-18	20-28	5.5-8.0	11-16	Raises dust and loose paper; small branches are moved.	Moderate Breeze	
5	19-24	29-38	8.0-10.8	17-21	Small trees in leaf begin to sway; crested wavelets form on inland waters.	Fresh Breeze	Reduce kite size increase line weight & drogue size
6	25-31	39-49	10.8-13.9	22-27	Large branches in motion; whistling heard in telegraph.	Strong Breeze	
7	32-38	50-61	13.9-17.2	28-33	Whole trees in motion; inconvenience felt when walking.	Near Gale	KAP not possible without severe risk of injury to operator and equipment.
8	39-46	62-74	17.2-20.7	34-40	Breaks twigs off trees; generally impedes progress.	Gale	
9	47-54	75-88	20.7-24.5	41-47	Slight structural damage occurs (chimney-pots and slates removed).	Severe Gale	
10	55-63	89-102	24.5-28.4	48-55	Seldom experienced inland; trees uprooted; considerable structural damage occurs.	Storm	
11	64-72	103-117	28.4-32.6	56-63	Very rarely experienced; accompanied by wide-spread damage.	Violent Storm	
12	73-83	≥118	≥32.6	64-71		Hurricane	

This wind speed level is accurately referred to Beaufort scale as in Table 4.2. To calculate speed, distance travelled in a certain amount of time which is 0.67 m/s. Measuring time is fairly straightforward by count the number of signals over the course of a fixed time period. The result is tested by doing a comparison with the wind speed to Mastech MS2625B digital anemometer. This device is a portable and professional wind speed measuring instrument with large-screen LCD backlight and multi-unit switching. Its resolution of wind speed is 0.01 m/s. Its accuracy of wind speed measurement based on it percentage error in unit of m/s (meter per second) is ±(2.0%+50) for measuring range of 0.80 to 30.00 m/s. Strategically, method as shown in Figure 4.8 can verify this weather station accurateness of wind speed.

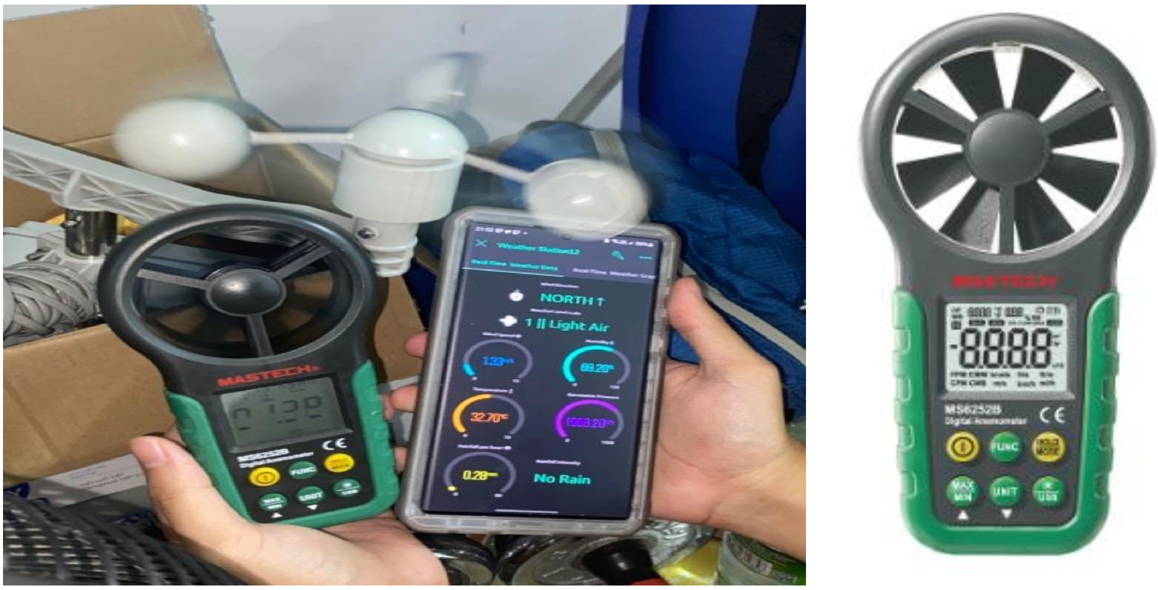


Figure 4.8 Verify project wind speed against Mastech digital anemometer

Table 4.4: Data wind speed Weather station against digital anemometer

Wind Speed Home Weather Station	Wind Speed Digital Anemometer	Beaufort Force level	Accuracy (Percentage Error)
0.67 m/s	0.68 m/s	1	±(1.47%)
1.33 m/s	1.39 m/s	1	±(4.31%)
2.00 m/s	2.07 m/s	2	±(3.38%)
2.68 m/s	2.73 m/s	2	±(1.83%)
3.35 m/s	3.43 m/s	3	±(2.33%)
4.00 m/s	3.94 m/s	3	±(1.52%)

Weather station is calibrated by testing it with certain wind speed level in certain period. The result is then being compared to wind speed from digital anemometer with the same wind speed level. This is continued with the different wind speed level. Each wind speed reading is based on Beaufort scale to determine its force level. By then, wind speed data between it can be used to calculate percentage error of weather station wind speed measurement. Table 4.4 shows data wind speed weather station result against digital anemometer. Based on it, percentage error didn't exceed more than 5% so this weather station can be verified its accurateness and reliability on measuring wind speed. It also resulting in the same Beaufort force level for every recorded measurement.

4.3.5 Durability of hardware design

This weather station is designed and targeted to be robust and durable especially in outdoor, during extreme weather or at high-altitude place. A weather station main functional is for collecting local climate and environmental data. Once, its running and tested, it needs to be robust in build so that can deploy it outside and it will be reliable in the long term. This weather station needs to remain operational even in blustery conditions as unexpected weather can occur anytime. To add with its durability against such weather, these anemometer, wind vane and rain gauge device has come with RJ11 connectors which are sturdy and therefore difficult to accidentally dislodge, so weather station will remain operational even in blustery conditions. For a more robust, long-term installation, or to save room from size of breadboard in enclosure, electronics components with PCB board inside can be prevent from get wet or even very damp with solid durability of weather station. If such wet or damp condition happens inside the enclosure to the components, the possibility they will fail or start to corrode might occurs. Weatherproof enclosure to house the external environmental sensors could solve this that’s why it is implement in this project. This durability is support by enclosure that place all electronic components inside which is rate as IP56. The IP (Ingress Protection) enclosure system uses a 2-digit suffix to describe the degree of ingress protection for enclosures. The IP rating is an international standard. It determines the degree of protection or sealing efficacy in enclosures against the infiltration of objects, water, dust, and contact. Also, it is compliant with European Standard EN 60529. In this project, the enclosure is rate IP56 and offers deeper protection, ensuring powerful, high-flow jets of water also won’t get in. IP56 rating, indicating it can withstand a blast from a pressure washer at a reasonable distance and is well protected from dust. Also providing a better option near water. Figure 4.9 shows weather station able to withstand in outdoor during rainy and windy weather.

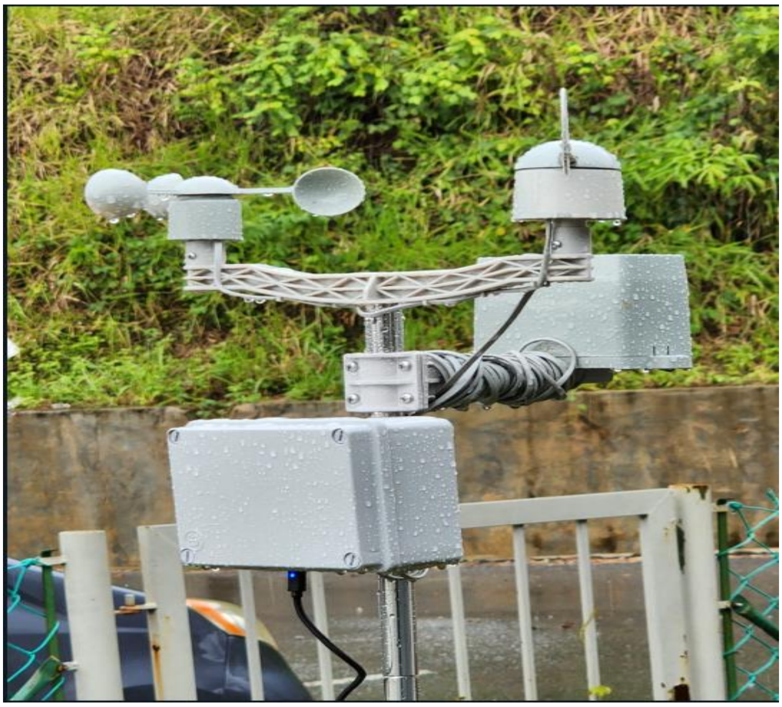


Figure 4.9 Weather station running outdoor in rainy and windy weather

4.4 Weather station data analysis

Gathering, analyzing, and storing data from this weather station is tested on January 4, 2023. Analysis can be done after weather station has been tested in whole day to observe all the sensors individually. By adapt and integrate the code that have been written for to regularly measure and record data, it can be set to observe for whole day, week, month or even a year. This overall data can be monitor through Blynk app to record the complete data collection system. Blynk app also able to display graph based on data collected and classified it accordingly based on hour, day, week, month, and year. This data analysis is critical if minor changes to weather parameters data are required. Although the calibration of sensor has been done and verified, testing the whole system running 24 hours outdoor is another story. Simulation of wind activity outdoor by installing weather station outdoor to keep the rotating vane and anemometer so its measurements is based on real time data. It can give the raw data to process on its own. Within Wi-Fi connectivity, live data can easily be access through Blynk app every time weather station update its weather data regularly in a second.

4.4.1 Weather station data analysis against weather forecast website

Overall weather data for wind parameter, temperature, humidity, and barometric pressure are compared to the Zoom Earth website which is a trusted and reliable web for real time weather data because its data is provides by satellite imagery. It’s a reliable source to use as reference for this weather station data. Real time images are sourced from NASA satellites and are updated every day from time to time. It’s recorded the data even in minutes for every weather parameter. Zoom Earth can be described as visualizes global weather in real-time. It can also track hurricanes, tropical storms, severe weather, wildfire smoke and much more. In addition, all its data can be monitor through “Watch LIVE” satellite images with the latest weather radar. Its user interface is also pleasant and user friendly as it features such beautiful interactive weather forecast maps of rain, snow, wind speed, temperature, humidity, and pressure. This is a perfect website to match all this weather station data as the data needed is specifically in one day only. In a day, the data for each weather parameter is recorded during 7:00, 19:00 and 23:50 respectively. Based on these 3 specific times of recorded period, this weather station data can be verified, gathered, analyzed, stored for its preciseness. The tolerance of error in data can be minimized to avoid unnecessary parameter to be recorded. Although there is some issue with humidity and temperature data that produced contradictory readings because both of this parameter is inside the same sensor which must be keep inside with other sensors part since the enclosure to store the sensors is a bit dense inside the box. This is a bit misjudges in prototype design because accuracy has been sacrificed in favor of ease of use to keep the electronic device waterproof and safe from such rainy weather that could damage components inside enclosure. Overall, the comparison result is quite satisfying since most of the sensor for this weather station able to get almost the same result as the satellite imagery from Zoom earth website although there are some factors need to be considered such as specific location since Zoom earth website only recorded a nearest location based on district of Alor Gajah as this personal weather station is tested in Taman Belimbing Harmoni, Durian Tunggal region. Also, some other factor such as placement or mounting of installed weather station which is not on the roof or highest location which is more precise way to record wind speed in general. These are the result comparison for weather station data in Blynk app against data on zoom earth website on January 4,2023 shown as Figure 4.10, 4.11, 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19.

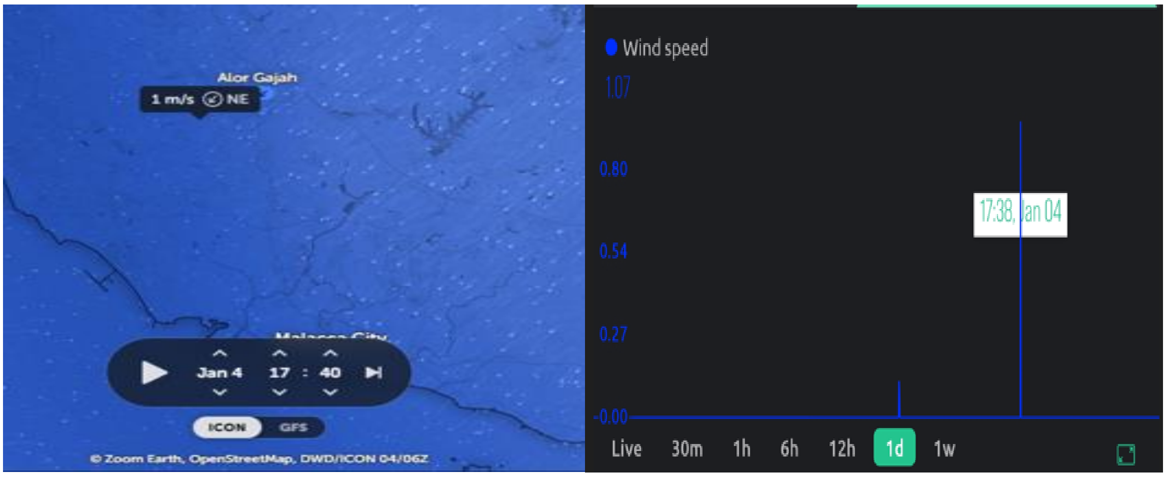


Figure 4.10 Wind speed is recorded 1 m/s and direction is Northeast during 17:40

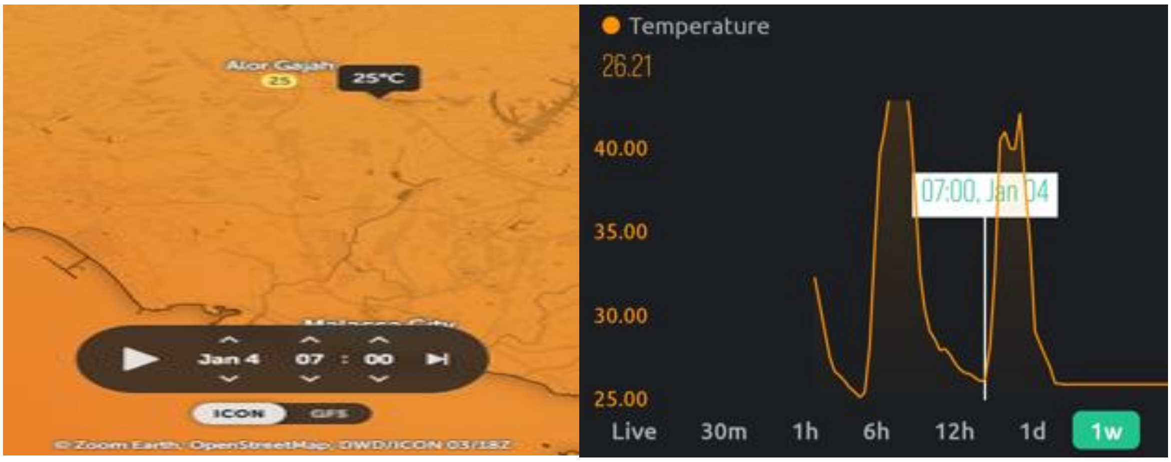


Figure 4.11 Temperature during 7:00

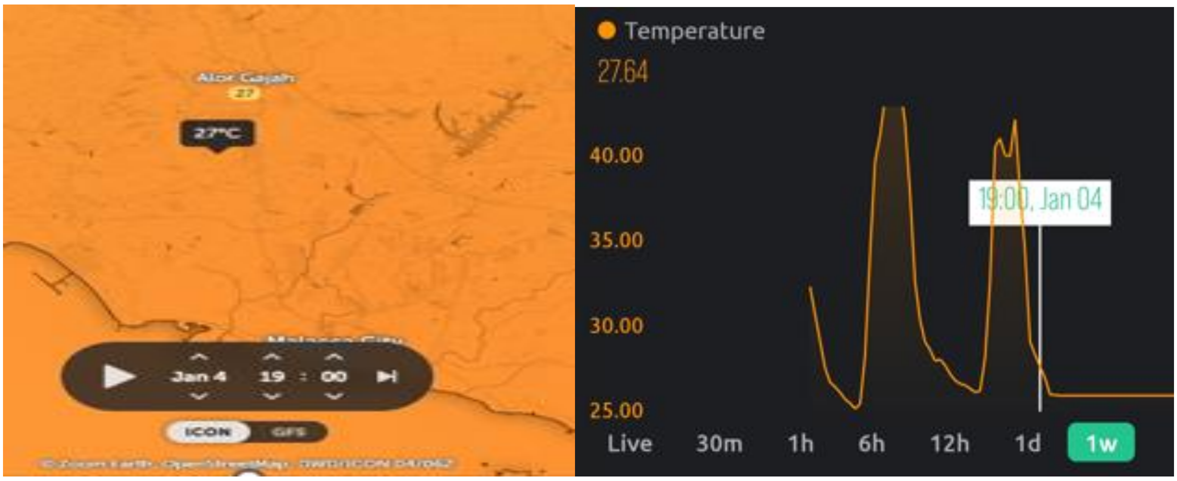


Figure 4.12 Temperature during 19:00

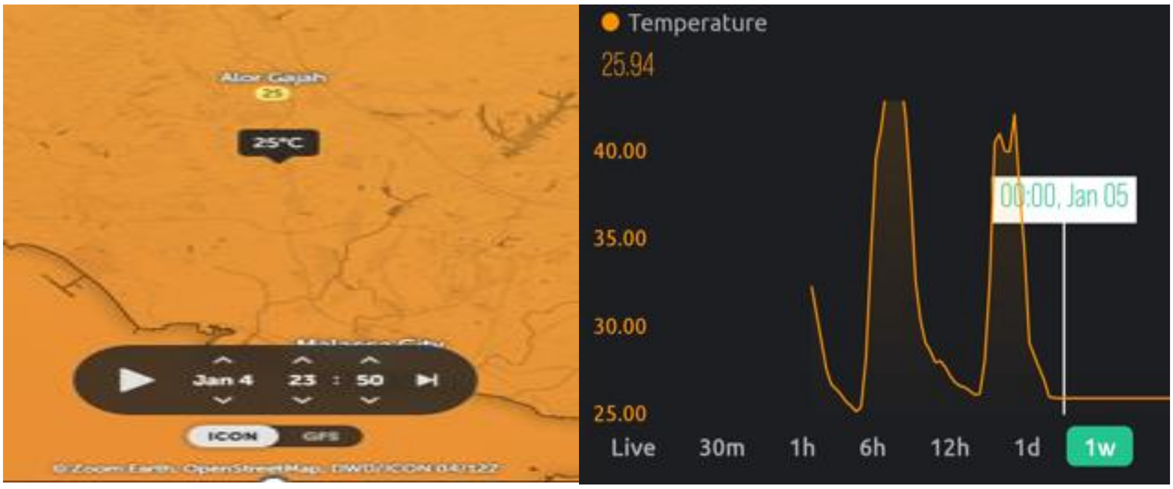


Figure 4.13 Temperature during 23:50

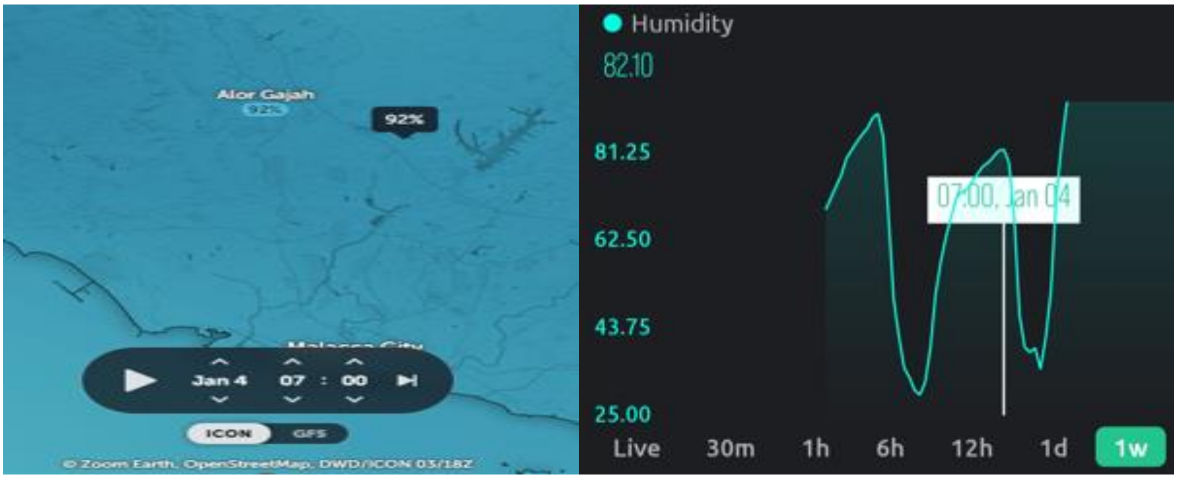


Figure 4.14 Humidity during 7:00

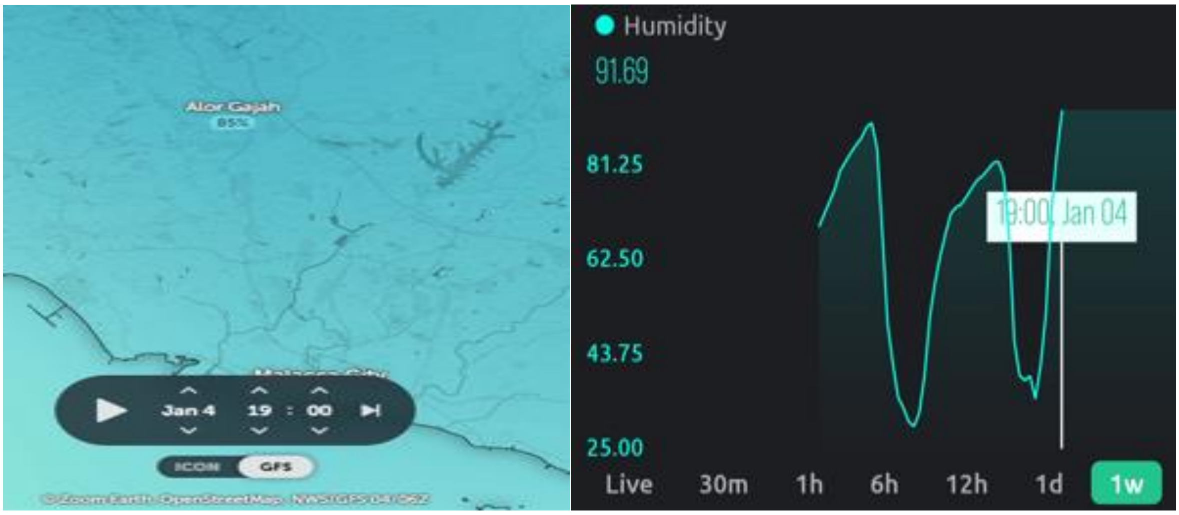


Figure 4.15 Humidity during 19:00

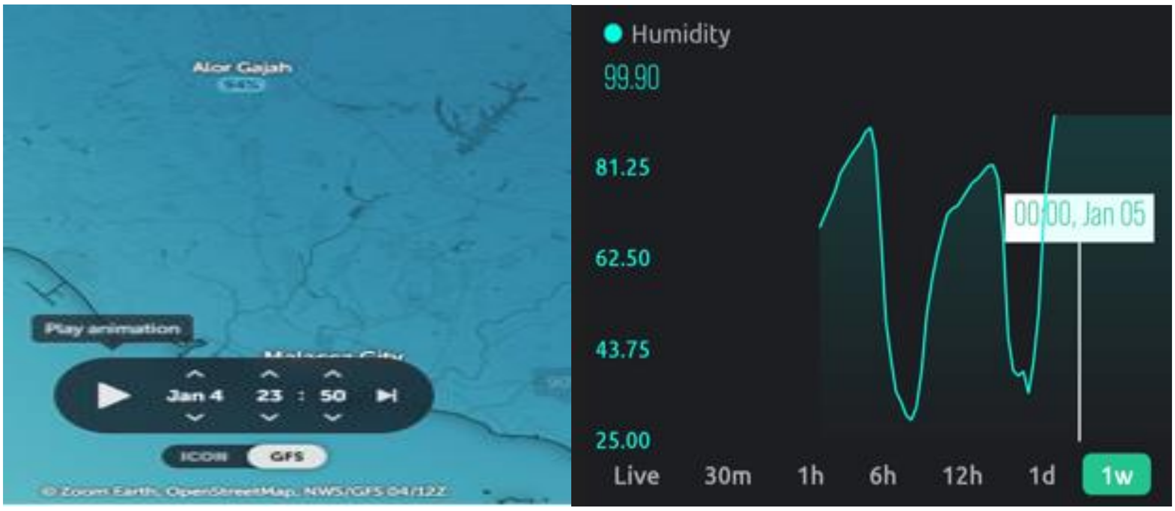


Figure 4.16 Humidity during 23:50

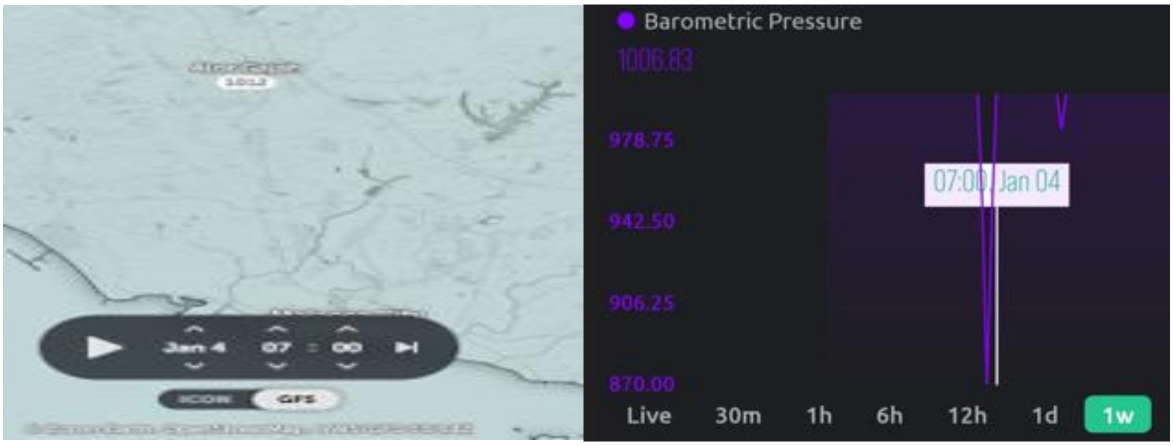


Figure 4.17 Barometric pressure during 7:00

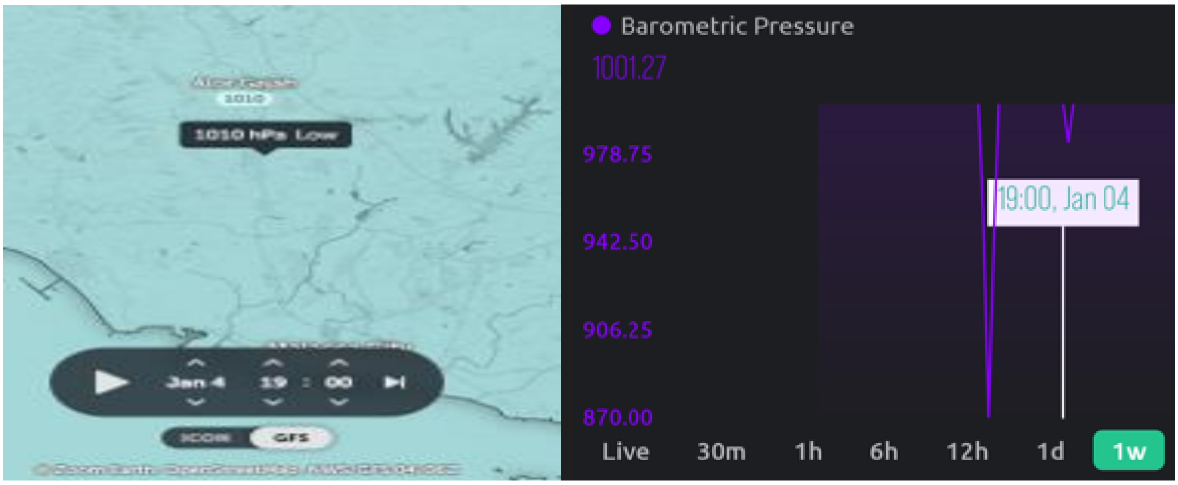


Figure 4.18 Barometric pressure during 19:00

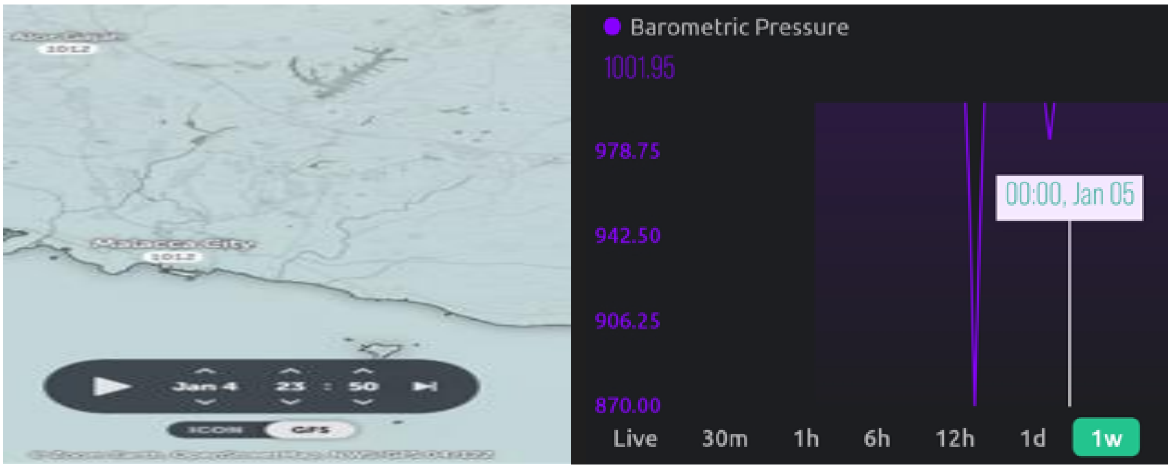
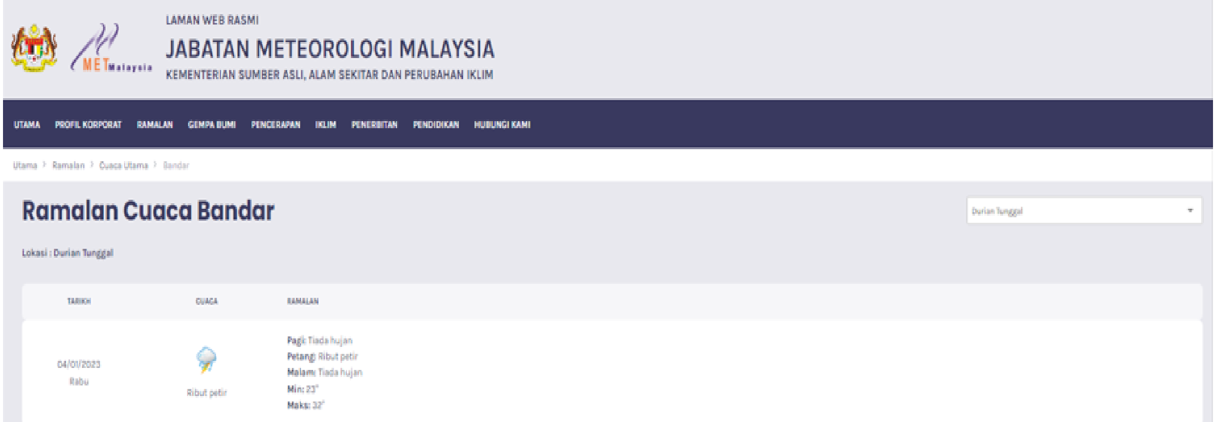


Figure 4.19 Barometric pressure during 23:50

The overall result also compared to forecast from Malaysian Meteorological Department as shown in Figure 4.20. The data is filtered to region of installed weather station which is in district of Durian Tunggal. The max temperature is 32°C while its min temperature is 23°C and it is predicted to be rain on evening. Figure 4.21 shows graph result of weather data.


LAMAN WEB RASMI

JABATAN METEOROLOGI MALAYSIA

KEMENTERIAN SUMBER ASLI, ALAM SEKITAR DAN PERUBAHAN IKLIM

Ramalan Cuaca Bandar

Lokasi : Durian Tunggal

TARIKH	CUACA	RAMALAN
04/01/2023 Rabu	Ribut petir	Pagi: Tiada hujan Petang: Ribut petir Malam: Tiada hujan Min: 23° Maks: 32°

Figure 4.20 Weather forecast on January 4,2023 from Malaysian Meteorological

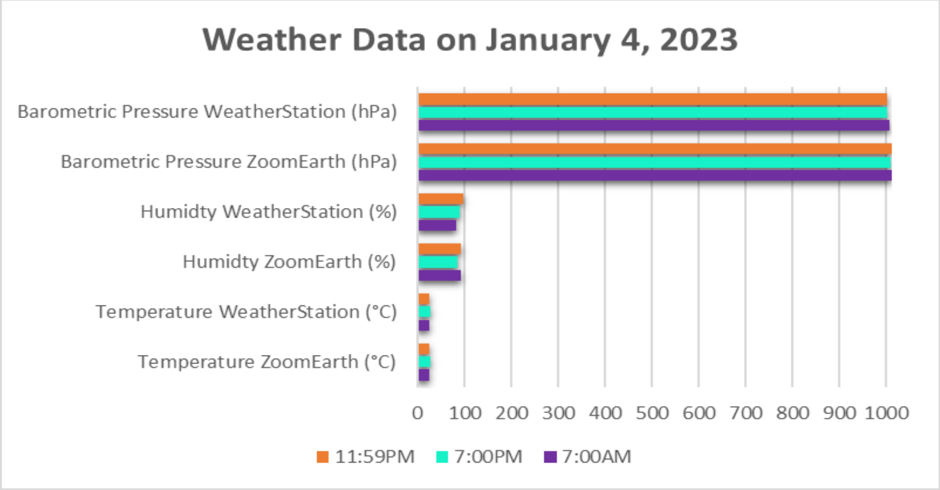
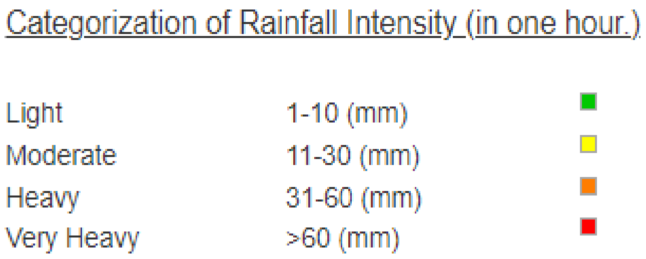


Figure 4.21 Overall graph result of weather data comparison

4.4.2 Rainfall data analysis based on public info banjir website

As for rainfall data, the method is quite different since it is based on the official web of Public InfoBanjir. The data to be observed is filter to Melaka state in Durian Tunggal district to get most accurate measurement since this weather station is installed in Taman Belimbing Harmoni, Durian Tunggal. According to data from that website, convective rain more than 60 mm in 2 to 4 hours duration (typical) may cause flash floods. However, monsoon rains are typically of long duration with intermittent heavy bursts and the intensity can occasionally exceed several hundred mm in 24 hours. In this case, rainfall intensity categorization data is based on data by Public InfoBanjir as shown in Figure 4.22. Based on that, 1-10mm rainfall is categorized as light rain, 11-30mm rainfall is categorized as moderate rain, 31-60mm is categorized heavy rain and more than 60mm is categorized as very heavy rain. This rainfall intensity categorization is then displayed on the Blynk app in accordance with its classification.

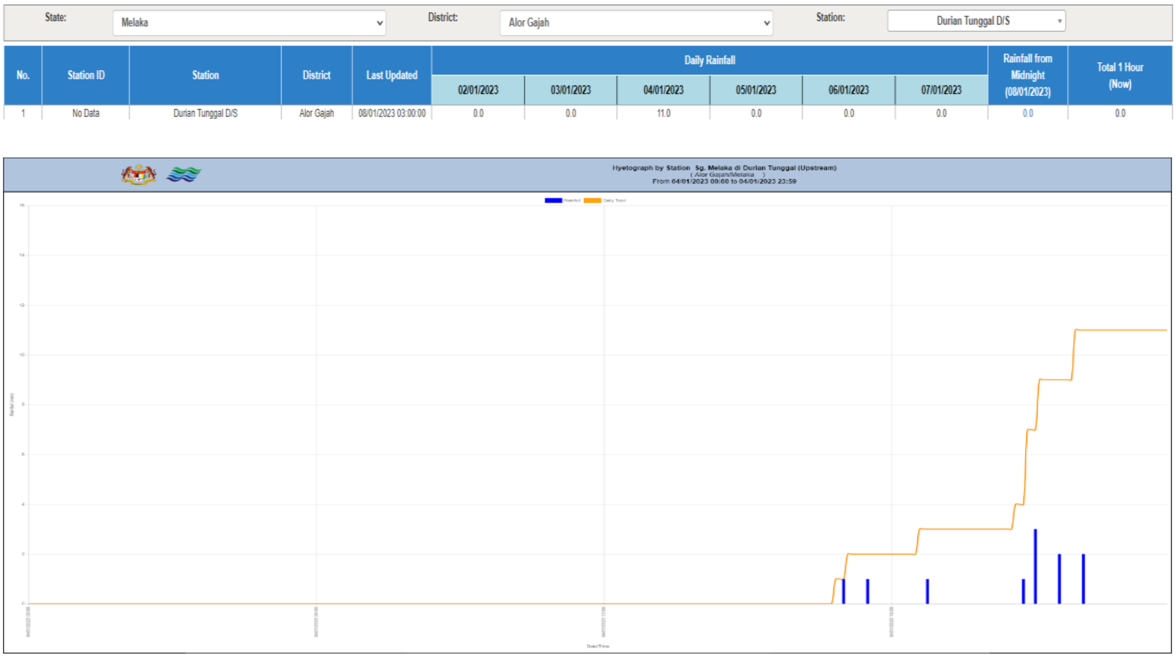


Categorization of Rainfall Intensity (in one hour.)

Light	1-10 (mm)	🟩
Moderate	11-30 (mm)	🟨
Heavy	31-60 (mm)	🟧
Very Heavy	>60 (mm)	🟥

Figure 4.22 Categorization of Rainfall Intensity

Upon testing the weather station on January 4, 2023, rainfall start accumulating in evening around 17:40 with rainfall intensity of 1.68mm categorized as light rain. Around 19:13, rainfall start increasing in its value to 3.07mm. Then, rainfall intensity of 10.34mm is at peak categorized as moderate rain around 21:35. The value is assumed as 11mm although weather station measured 10.34mm because of some contradictory readings from rainfall sensor. Such factor of roof, wall and tree blocking the rainfall, or the rain gauge is partially sheltered by an overhang, and inaccuracy of location of installed weather station is taking into consideration for this minor error. These overall data from weather station is then compared to Public InfoBanjir website recorded data to analyze the accurateness of sensor. Overall, rainfall data from weather station is quite good since its almost tally against rainfall data from Public InfoBanjir website. Figure 4.23 and Figure 4.24 shows rainfall data comparison between Public InfoBanjir website and weather station. Figure 4.25 shows graph result for this comparison.



State: Melaka District: Alor Gajah Station: Durian Tunggal D/S

No.	Station ID	Station	District	Last Updated	Daily Rainfall						Rainfall from Midnight (08/01/2023)	Total 1 Hour (Now)
					02/01/2023	03/01/2023	04/01/2023	05/01/2023	06/01/2023	07/01/2023		
1	No Data	Durian Tunggal D/S	Alor Gajah	08/01/2023 03:00:00	0.0	0.0	11.0	0.0	0.0	0.0	0.0	0.0

Hyetograph by Station Sg. Melaka di Durian Tunggal (Upstream)
(Alor Gajah/Melaka)
From 04/01/2023 00:00 to 04/01/2023 23:59
Rainfall
Daily Total

Figure 4.23 Public InfoBanjir website rainfall data on January 4, 2023

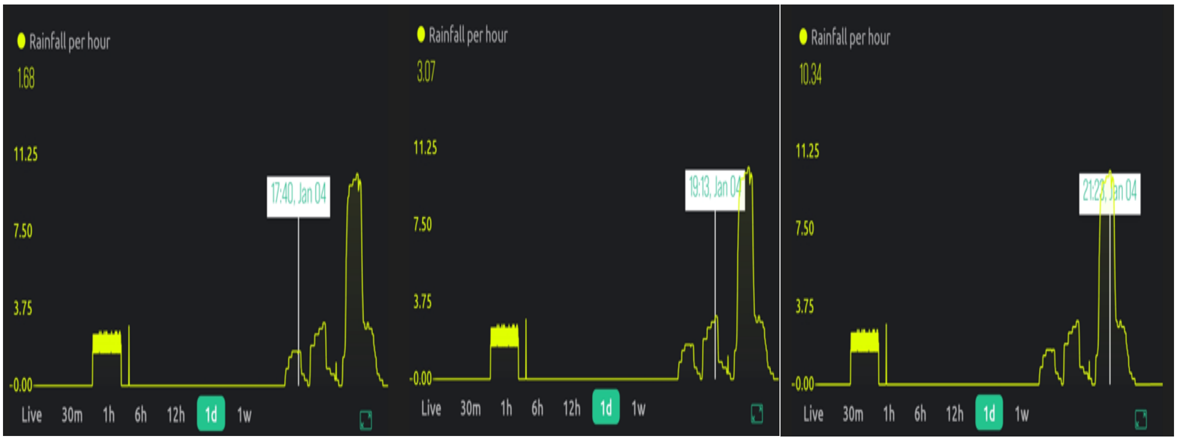


Figure 4.24 Rainfall data on weather station for 17:40, 19:13 and 21:23

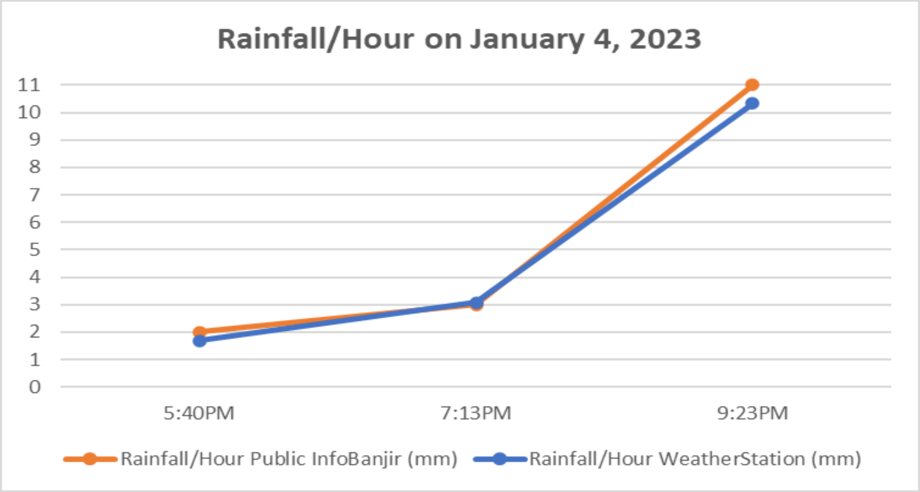


Figure 4.25 Graph result of rainfall data comparison

4.5 Summary

This based home weather station equipment is a product which is very useful to assist the weather enthusiast who want to know about real time weather data or people who curious about rapid weather changes in their location. Weather data are very useful for evaluation and comparison of climatic changes and effects of global warming. This implementation able to produce higher accuracy and stability measurement value. In addition, it provides clean app interface which will be useful for user to monitor the weather data.

CHAPTER 5

CONCLUSION

5.1 Introduction

This chapter summarizes the entire development process of based home weather station equipment to measure wind speed and direction. Furthermore, this chapter also provide suggestion and recommendation to improve and develop the based home weather station equipment to become more reliable to use in household and as personal weather station.

5.2 Conclusion

The government and commercial organizations in our region typically construct weather stations in remote locations, making it difficult for average citizens to easily access the meteorological information they provide. There are many apps in android to check the weather data in our present location. Although the weather data may be true to some extent, the distance between the weather station and your current location can significantly alter the readings. For example, a weather station that installed in nearer to a seashore is more accurate than weather station that installed far away from seashore then there will be big changes in wind flow comparing in both regions. Based Home Weather Station Equipment is designed with flexible system configuration and hardware portability which means it easy to install in any circumstance as an example equip at such a high place and no need to worry about the hardware since all the components will be merge into PCB layout hence make it a market-ready product. This equipment is designed to use in any household area but the advantage in windy area or high-altitude place such ideal location like seashore or mountain since the wind parameter can be measure with more precisely.

In conclusion, lot of time and effort were put on this project in order to design an initial performance of based home weather station equipment. The milestones are created in orderly to achieve the objectives of this project as a guidance for the project undergoes. Hence, all the objectives were achieved since milestones are constructed in orderly which guided the overall system development and the process of data analysis. Inside each milestone, the process to complete the task was explained clearly by using flowchart method. The architecture design of this based home weather station equipment was designed with wind vane and 3 cup anemometer to observe and measure wind speed and direction parameter. Wind speed is measure in meter per second (m/s) while wind direction in 8 pole direction which are North, North East, East, South East, South, South West, West and North West. After that, these data will be sent to cloud and can be directly observe through internet connectivity through smartphone with Blynk app. In the meantime, prototype was designed in 3D TinkerCAD to give early impression and analysis about the mechanism of this weather station. On top of that, this weather station design able to keep all its electronic components watertight, thus able prevent any rainfall enter the enclosure in any harmful quantity. At last, this project design has successfully able to make weather station up and running outdoor to measure all of weather parameter data through Wi-Fi connectivity by Blynk app.

5.3 Recommendation

There are always a room of improvement for the system designed and the areas that can be further improvised are as follows:

i. A wind gust is a brief increase in wind speed that can occur whenever the wind is blowing. Add functionality in system to measure wind gust.

ii. Put several new sensors such feature to detect lightning, UV sensor, light meter, and soil moisture in addition to weather station.

iii. Develop a weather station that fully depends on solar power as its main power source.

iv. Add functionality to increase wind direction to 16 possible directions as the system can only provide 8 directions.

REFERENCES

[1] Abu Saleh Bin Shahadat, Safial Islam Ayon, Most. Rokeya Khatun. "Efficient IoT based Weather Station" , 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE), 2020

[2] Palak Kapoor, Ferdous Ahmed Barbhuiya. "Cloud Based Weather Station using IoT Devices" , TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), 2019

[3] Laszlo-Zsolt Turos, Geza Csernath, Barna Csenteri. "Power Management In IoT Weather Station" , 2018 International Conference and Exposition on Electrical And Power Engineering (EPE), 2018

[4] Norakmar binti Arbain Sulaiman, Muhamad Dan Darrawi bin Sadli. "An IoT-based Smart Garden with Weather Station System" , 2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2019

[5] Md. Jahirul Alam, Shoyeb Ahammad Rafi, Ali Adnan Badhan, Md. Najmul Islam, Saiful Islam Shuvo, Ahmed Mortuza Saleque. "Low Cost IoT Based Weather Station for Real-Time Monitoring" , 2020 IEEE 2nd International Conference on Circuits and Systems (ICCS), 2020

[6] Medilla Kusriyanto, Agusti Anggara Putra. "Weather Station Design Using IoT Platform Based On Arduino Mega" , 2018 International Symposium on Electronics and Smart Devices (ISESD), 2018

[7] Medilla Kusriyanto, Agusti Anggara Putra. "Weather Station Design Using IoT Platform Based On Arduino Mega" , 2018 International Symposium on Electronics and Smart Devices (ISESD), 2018

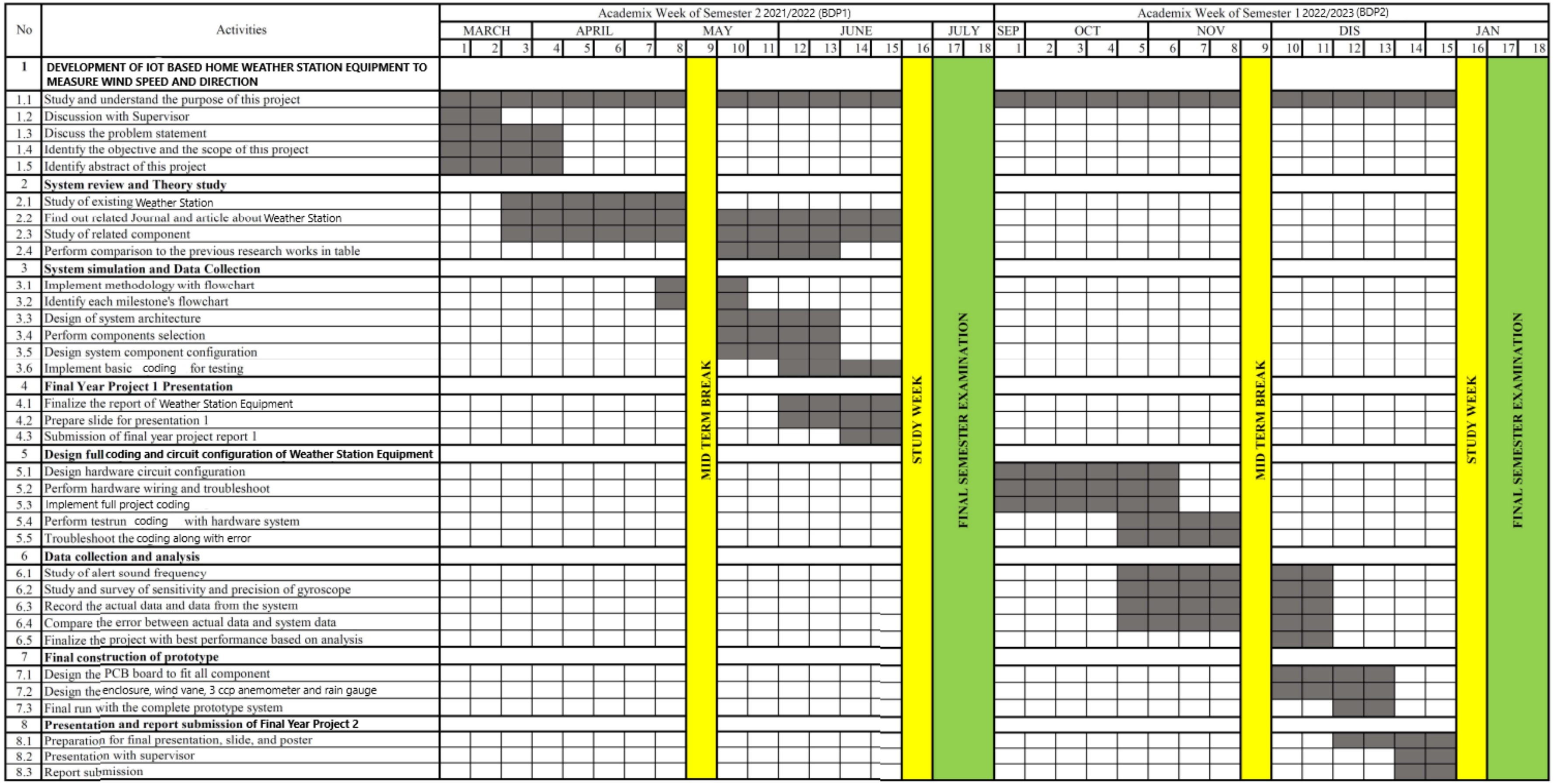
[8] Anang Suryana, Fitra Phila Lismana, Rizky Maulana Rachmat, Septian Dwi Putra, Marina Artiyasa. "Implementation of Weather Station for The Weather Reality In A Room" , 2020 6th International Conference on Computing Engineering and Design (ICCED), 2020

[9] Padma Balaji Leelavinodhan, Massimo Vecchio, Fabio Antonelli, Andrea Maestrini, Davide Brunelli. "Design and Implementation of an Energy-Efficient Weather Station for Wind Data Collection" , Sensors, 2021

[10] Kriddikorn Kaewwongsri, Kittasil Silanon. "Design and Implement of a Weather Monitoring Station using CoAP on NB-IoT Network" , 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2020

APPENDICES

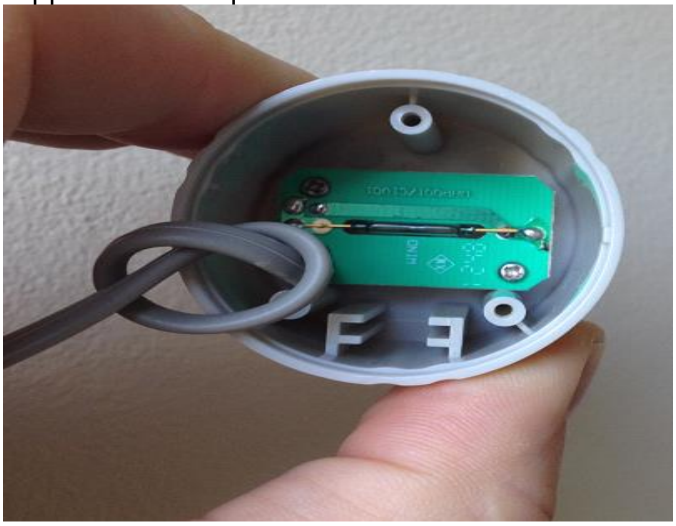
Appendix A Gantt Chart



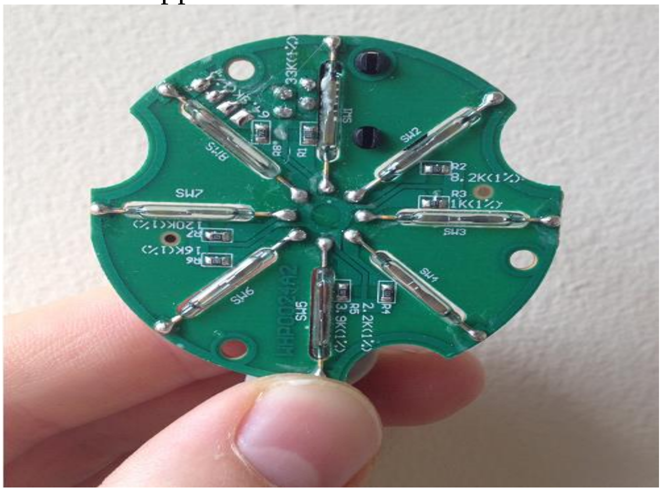
No	Activities	Academix Week of Semester 2 2021/2022 (BDP1)																		Academix Week of Semester 1 2022/2023 (BDP2)																	
		MARCH			APRIL				MAY				JUNE					JULY		SEP	OCT				NOV				DIS					JAN			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	**DEVELOPMENT OF IOT BASED HOME WEATHER STATION EQUIPMENT TO MEASURE WIND SPEED AND DIRECTION**									MID TERM BREAK							STUDY WEEK	FINAL SEMESTER EXAMINATION										MID TERM BREAK							STUDY WEEK	FINAL SEMESTER EXAMINATION	
1.1	Study and understand the purpose of this project	■	■	■	■	■	■	■	■		■	■	■	■	■	■				■	■	■	■	■	■	■	■		■	■	■	■	■	■			
1.2	Discussion with Supervisor	■	■																																		
1.3	Discuss the problem statement	■	■	■	■																																
1.4	Identify the objective and the scope of this project	■	■	■	■																																
1.5	Identify abstract of this project	■	■	■	■																																
2	**System review and Theory study**																																				
2.1	Study of existing Weather Station			■	■	■	■	■	■																												
2.2	Find out related Journal and article about Weather Station			■	■	■	■	■	■		■	■	■	■	■	■																					
2.3	Study of related component			■	■	■	■	■	■		■	■	■	■	■	■																					
2.4	Perform comparison to the previous research works in table										■	■	■	■																							
3	**System simulation and Data Collection**																																				
3.1	Implement methodology with flowchart								■		■																										
3.2	Identify each milestone's flowchart								■		■																										
3.3	Design of system architecture										■	■	■	■																							
3.4	Perform components selection										■	■	■	■																							
3.5	Design system component configuration										■	■	■	■																							
3.6	Implement basic coding for testing												■	■	■	■																					
4	**Final Year Project 1 Presentation**																																				
4.1	Finalize the report of Weather Station Equipment												■	■	■	■																					
4.2	Prepare slide for presentation 1												■	■	■	■																					
4.3	Submission of final year project report 1														■	■																					
5	**Design full coding and circuit configuration of Weather Station Equipment**																																				
5.1	Design hardware circuit configuration																			■	■	■	■	■	■												
5.2	Perform hardware wiring and troubleshoot																			■	■	■	■	■	■												
5.3	Implement full project coding																			■	■	■	■	■	■												
5.4	Perform testrun coding with hardware system																							■	■	■	■										
5.5	Troubleshoot the coding along with error																							■	■	■	■										
6	**Data collection and analysis**																																				
6.1	Study of alert sound frequency																							■	■	■	■		■	■							
6.2	Study and survey of sensitivity and precision of gyroscope																							■	■	■	■		■	■							
6.3	Record the actual data and data from the system																							■	■	■	■		■	■							
6.4	Compare the error between actual data and system data																							■	■	■	■		■	■							
6.5	Finalize the project with best performance based on analysis																												■	■							
7	**Final construction of prototype**																																				
7.1	Design the PCB board to fit all component																												■	■	■	■					
7.2	Design the enclosure, wind vane, 3 ccp anemometer and rain gauge																												■	■	■	■					
7.3	Final run with the complete prototype system																														■	■					
8	**Presentation and report submission of Final Year Project 2**																																				
8.1	Preparation for final presentation, slide, and poster																														■	■	■	■			
8.2	Presentation with supervisor																																■	■			
8.3	Report submission																																■	■			

Appendix B Magnet underneath 3 Cup Anemometer

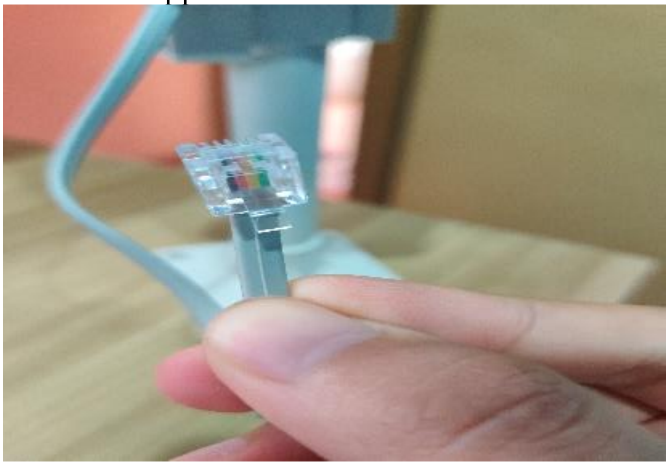
Appendix C3 Cup Anemometer reed switch PCB



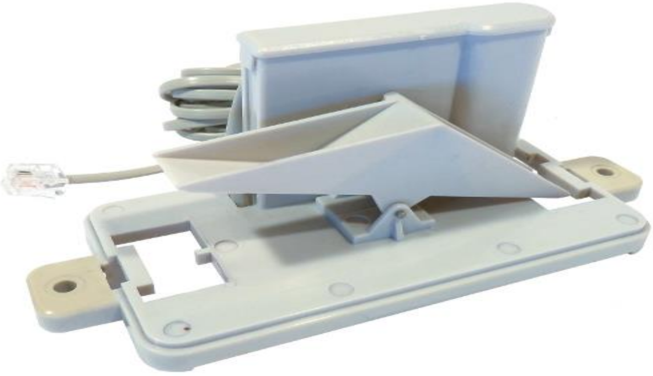
Appendix D Wind Vane PCB



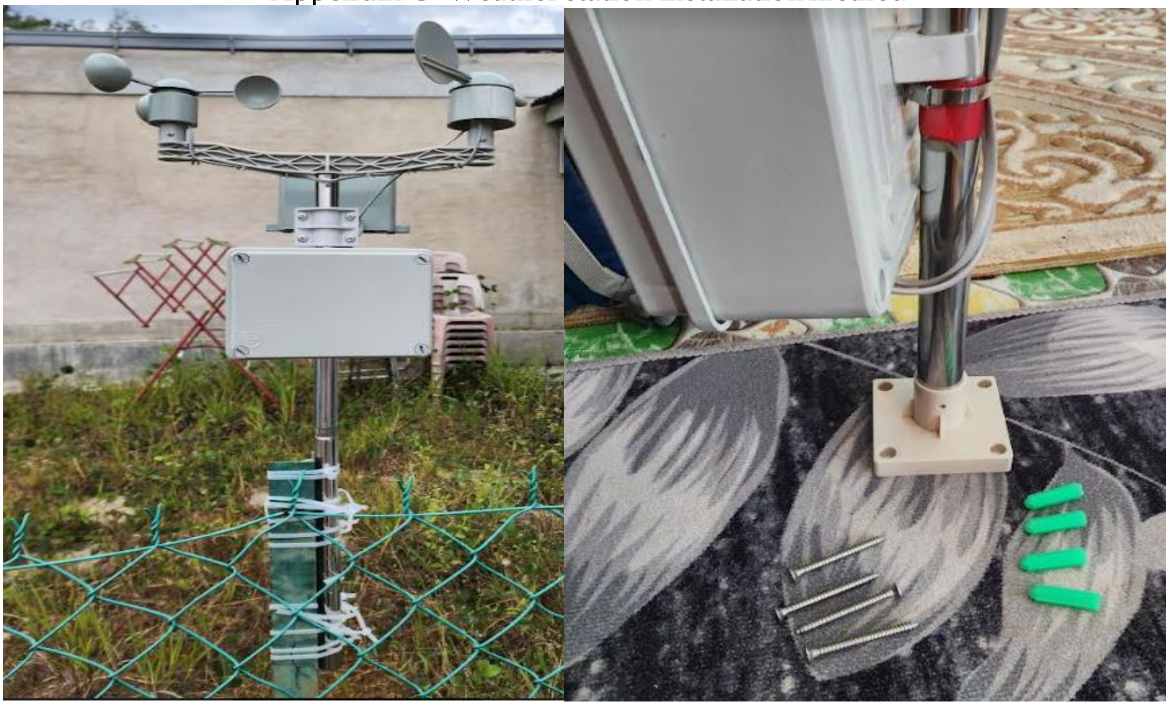
Appendix E RJ11 Connector



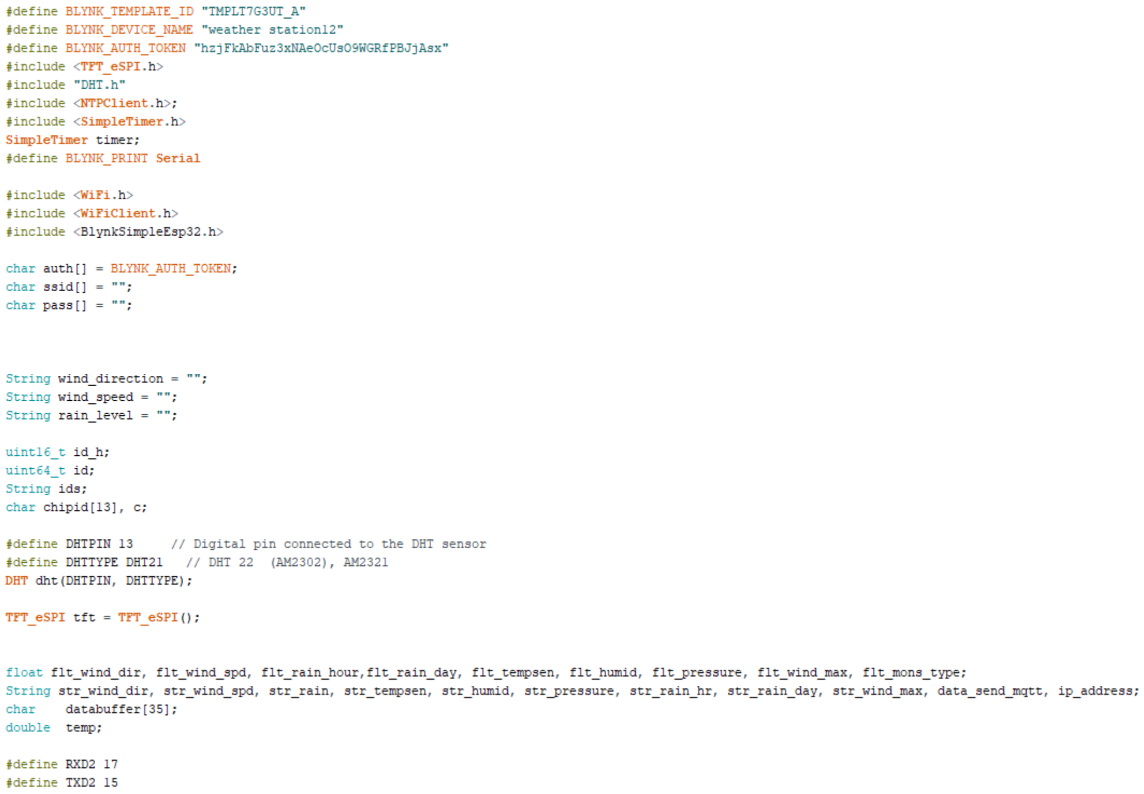
Appendix F Rain Gauge mechanism



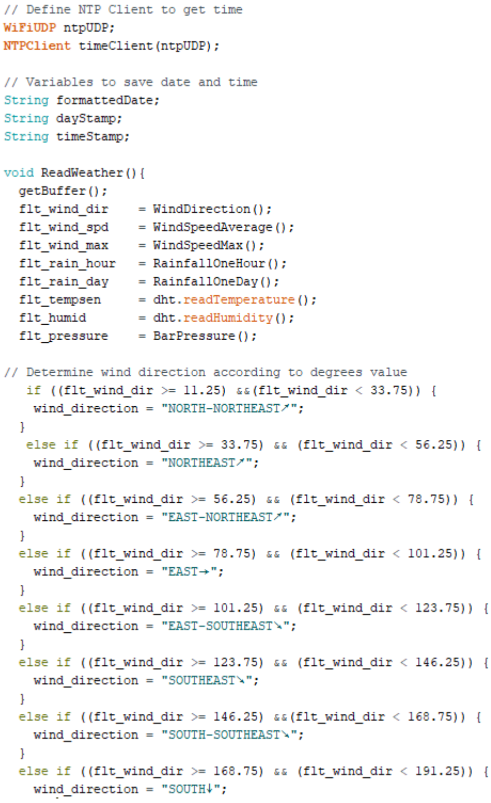
Appendix G Weather station installation method



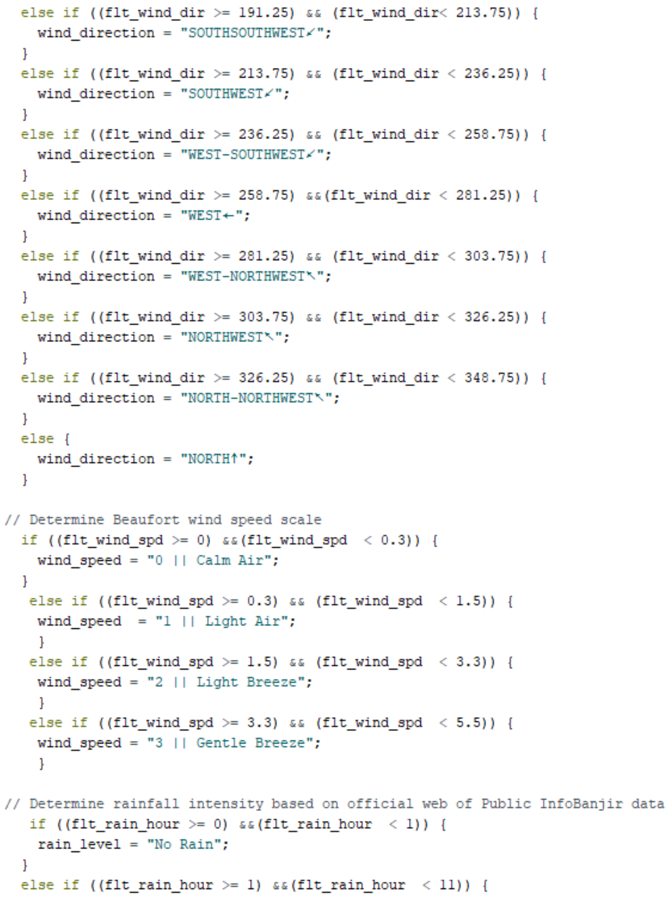
Appendix H Full Project Coding



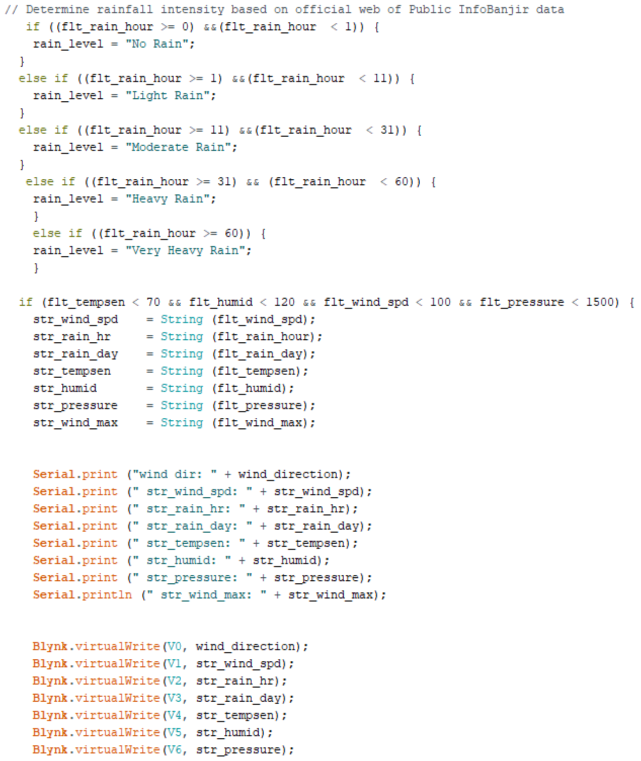
```
#define BLYNK_TEMPLATE_ID "TMPLT7G3UT_A"
#define BLYNK_DEVICE_NAME "weather station12"
#define BLYNK_AUTH_TOKEN "hzjFkAbFuz3xNAeOcUsO9WGRfPBJjAsx"
#include <TFT_eSPI.h>
#include "DHT.h"
#include <NTPClient.h>;
#include <SimpleTimer.h>
SimpleTimer timer;
#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "";
char pass[] = "";


String wind_direction = "";
String wind_speed = "";
String rain_level = "";

uint16_t id_h;
uint64_t id;
String ids;
char chipid[13], c;

#define DHTPIN 13     // Digital pin connected to the DHT sensor
#define DHTTYPE DHT21   // DHT 22  (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE);

TFT_eSPI tft = TFT_eSPI();


float flt_wind_dir, flt_wind_spd, flt_rain_hour,flt_rain_day, flt_tempsen, flt_humid, flt_pressure, flt_wind_max, flt_mons_type;
String str_wind_dir, str_wind_spd, str_rain, str_tempsen, str_humid, str_pressure, str_rain_hr, str_rain_day, str_wind_max, data_send_mqtt, ip_address;
char    databuffer[35];
double  temp;

#define RXD2 17
#define TXD2 15
```



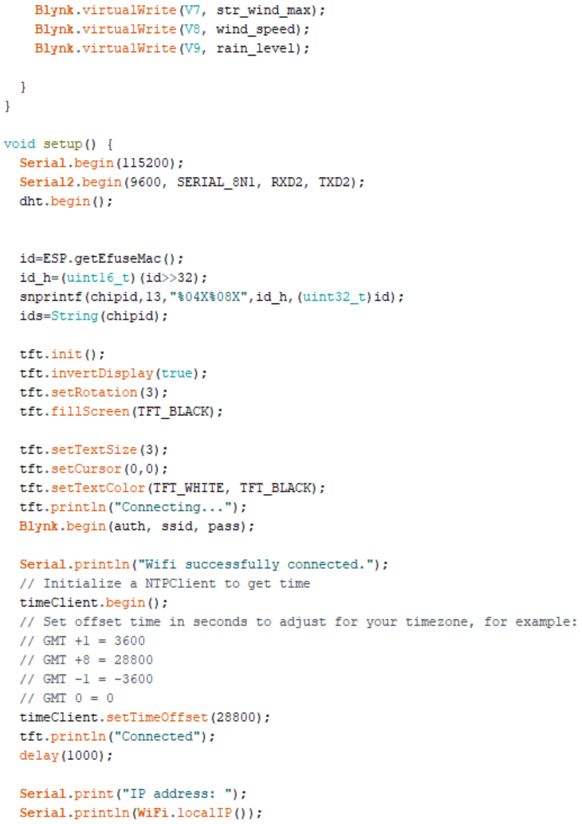
```
// Define NTP Client to get time
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP);

// Variables to save date and time
String formattedDate;
String dayStamp;
String timeStamp;

void ReadWeather(){
  getBuffer();
  flt_wind_dir     = WindDirection();
  flt_wind_spd     = WindSpeedAverage();
  flt_wind_max     = WindSpeedMax();
  flt_rain_hour    = RainfallOneHour();
  flt_rain_day     = RainfallOneDay();
  flt_tempsen      = dht.readTemperature();
  flt_humid        = dht.readHumidity();
  flt_pressure     = BarPressure();

// Determine wind direction according to degrees value
   if ((flt_wind_dir >= 11.25) &&(flt_wind_dir < 33.75)) {
    wind_direction = "NORTH-NORTHEAST↗";
  }
   else if ((flt_wind_dir >= 33.75) && (flt_wind_dir < 56.25)) {
    wind_direction = "NORTHEAST↗";
  }
  else if ((flt_wind_dir >= 56.25) && (flt_wind_dir < 78.75)) {
    wind_direction = "EAST-NORTHEAST↗";
  }
  else if ((flt_wind_dir >= 78.75) && (flt_wind_dir < 101.25)) {
    wind_direction = "EAST→";
  }
  else if ((flt_wind_dir >= 101.25) && (flt_wind_dir < 123.75)) {
    wind_direction = "EAST-SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 123.75) && (flt_wind_dir < 146.25)) {
    wind_direction = "SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 146.25) &&(flt_wind_dir < 168.75)) {
    wind_direction = "SOUTH-SOUTHEAST↘";
  }
  else if ((flt_wind_dir >= 168.75) && (flt_wind_dir < 191.25)) {
    wind_direction = "SOUTH↓";
```



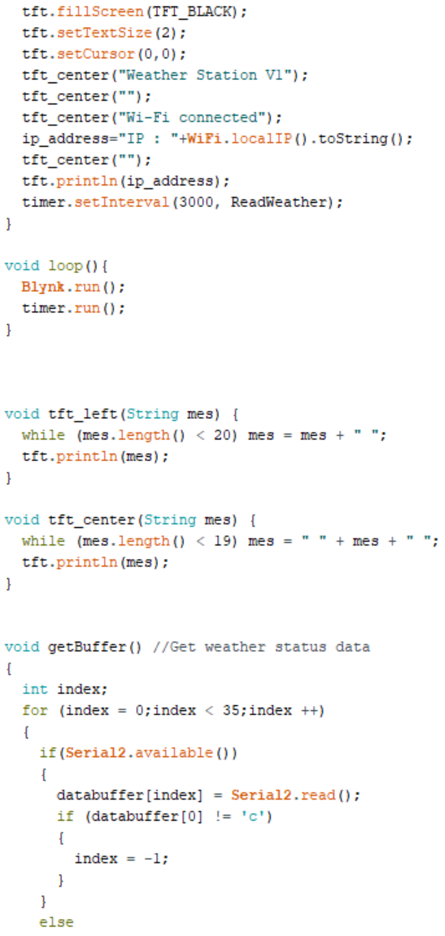
```
  else if ((flt_wind_dir >= 191.25) && (flt_wind_dir< 213.75)) {
    wind_direction = "SOUTHSOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 213.75) && (flt_wind_dir < 236.25)) {
    wind_direction = "SOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 236.25) && (flt_wind_dir < 258.75)) {
    wind_direction = "WEST-SOUTHWEST↙";
  }
  else if ((flt_wind_dir >= 258.75) &&(flt_wind_dir < 281.25)) {
    wind_direction = "WEST←";
  }
  else if ((flt_wind_dir >= 281.25) && (flt_wind_dir < 303.75)) {
    wind_direction = "WEST-NORTHWEST↖";
  }
  else if ((flt_wind_dir >= 303.75) && (flt_wind_dir < 326.25)) {
    wind_direction = "NORTHWEST↖";
  }
  else if ((flt_wind_dir >= 326.25) && (flt_wind_dir < 348.75)) {
    wind_direction = "NORTH-NORTHWEST↖";
  }
  else {
    wind_direction = "NORTH↑";
  }

// Determine Beaufort wind speed scale
  if ((flt_wind_spd >= 0) &&(flt_wind_spd  < 0.3)) {
    wind_speed = "0 || Calm Air";
  }
   else if ((flt_wind_spd >= 0.3) && (flt_wind_spd  < 1.5)) {
    wind_speed  = "1 || Light Air";
    }
   else if ((flt_wind_spd >= 1.5) && (flt_wind_spd  < 3.3)) {
    wind_speed = "2 || Light Breeze";
    }
   else if ((flt_wind_spd >= 3.3) && (flt_wind_spd  < 5.5)) {
    wind_speed = "3 || Gentle Breeze";
    }

// Determine rainfall intensity based on official web of Public InfoBanjir data
   if ((flt_rain_hour >= 0) &&(flt_rain_hour  < 1)) {
    rain_level = "No Rain";
  }
  else if ((flt_rain_hour >= 1) &&(flt_rain_hour  < 11)) {
```



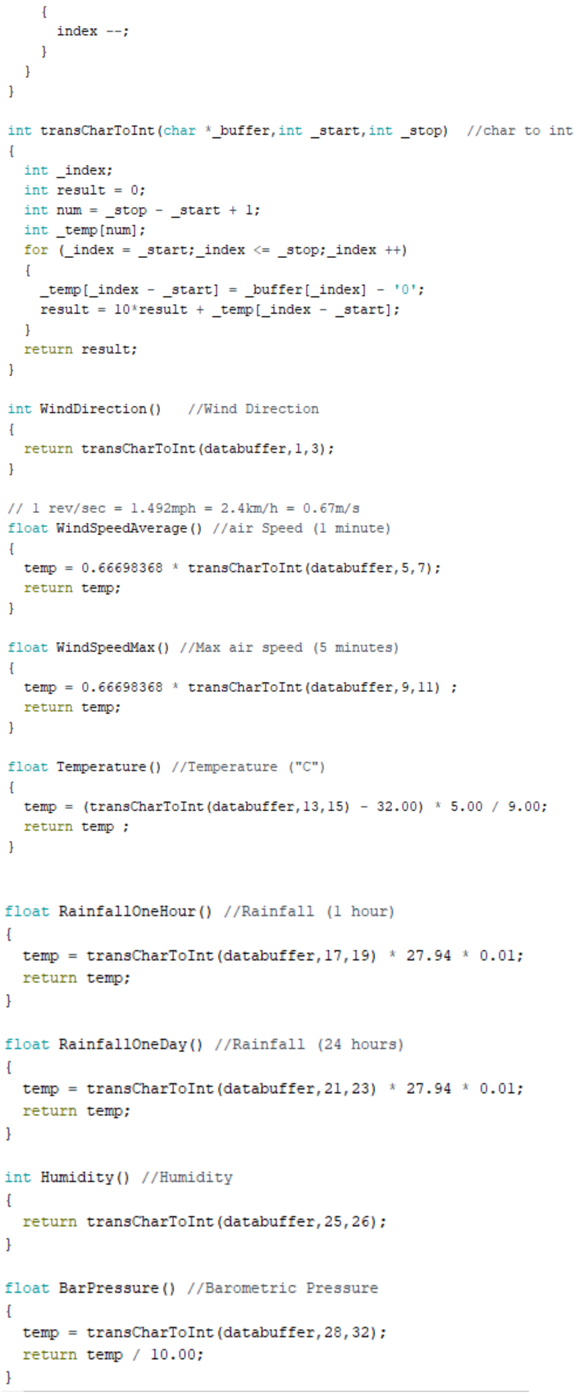
```
// Determine rainfall intensity based on official web of Public InfoBanjir data
   if ((flt_rain_hour >= 0) &&(flt_rain_hour  < 1)) {
    rain_level = "No Rain";
  }
  else if ((flt_rain_hour >= 1) &&(flt_rain_hour  < 11)) {
    rain_level = "Light Rain";
  }
  else if ((flt_rain_hour >= 11) &&(flt_rain_hour  < 31)) {
    rain_level = "Moderate Rain";
  }
   else if ((flt_rain_hour >= 31) && (flt_rain_hour  < 60)) {
    rain_level = "Heavy Rain";
    }
    else if ((flt_rain_hour >= 60)) {
    rain_level = "Very Heavy Rain";
    }

  if (flt_tempsen < 70 && flt_humid < 120 && flt_wind_spd < 100 && flt_pressure < 1500) {
    str_wind_spd     = String (flt_wind_spd);
    str_rain_hr      = String (flt_rain_hour);
    str_rain_day     = String (flt_rain_day);
    str_tempsen      = String (flt_tempsen);
    str_humid        = String (flt_humid);
    str_pressure     = String (flt_pressure);
    str_wind_max     = String (flt_wind_max);


    Serial.print ("wind dir: " + wind_direction);
    Serial.print (" str_wind_spd: " + str_wind_spd);
    Serial.print (" str_rain_hr: " + str_rain_hr);
    Serial.print (" str_rain_day: " + str_rain_day);
    Serial.print (" str_tempsen: " + str_tempsen);
    Serial.print (" str_humid: " + str_humid);
    Serial.print (" str_pressure: " + str_pressure);
    Serial.println (" str_wind_max: " + str_wind_max);


    Blynk.virtualWrite(V0, wind_direction);
    Blynk.virtualWrite(V1, str_wind_spd);
    Blynk.virtualWrite(V2, str_rain_hr);
    Blynk.virtualWrite(V3, str_rain_day);
    Blynk.virtualWrite(V4, str_tempsen);
    Blynk.virtualWrite(V5, str_humid);
    Blynk.virtualWrite(V6, str_pressure);
```



```
    Blynk.virtualWrite(V7, str_wind_max);
    Blynk.virtualWrite(V8, wind_speed);
    Blynk.virtualWrite(V9, rain_level);

  }
}

void setup() {
  Serial.begin(115200);
  Serial2.begin(9600, SERIAL_8N1, RXD2, TXD2);
  dht.begin();


  id=ESP.getEfuseMac();
  id_h=(uint16_t)(id>>32);
  snprintf(chipid,13,"%04X%08X",id_h,(uint32_t)id);
  ids=String(chipid);

  tft.init();
  tft.invertDisplay(true);
  tft.setRotation(3);
  tft.fillScreen(TFT_BLACK);

  tft.setTextSize(3);
  tft.setCursor(0,0);
  tft.setTextColor(TFT_WHITE, TFT_BLACK);
  tft.println("Connecting...");
  Blynk.begin(auth, ssid, pass);

  Serial.println("Wifi successfully connected.");
  // Initialize a NTPClient to get time
  timeClient.begin();
  // Set offset time in seconds to adjust for your timezone, for example:
  // GMT +1 = 3600
  // GMT +8 = 28800
  // GMT -1 = -3600
  // GMT 0 = 0
  timeClient.setTimeOffset(28800);
  tft.println("Connected");
  delay(1000);

  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
```



```
  tft.fillScreen(TFT_BLACK);
  tft.setTextSize(2);
  tft.setCursor(0,0);
  tft_center("Weather Station V1");
  tft_center("");
  tft_center("Wi-Fi connected");
  ip_address="IP : "+WiFi.localIP().toString();
  tft_center("");
  tft.println(ip_address);
  timer.setInterval(3000, ReadWeather);
}

void loop(){
  Blynk.run();
  timer.run();
}



void tft_left(String mes) {
  while (mes.length() < 20) mes = mes + " ";
  tft.println(mes);
}

void tft_center(String mes) {
  while (mes.length() < 19) mes = " " + mes + " ";
  tft.println(mes);
}


void getBuffer() //Get weather status data
{
  int index;
  for (index = 0;index < 35;index ++)
  {
    if(Serial2.available())
    {
      databuffer[index] = Serial2.read();
      if (databuffer[0] != 'c')
      {
        index = -1;
      }
    }
    else
```



```
    {
      index --;
    }
  }
}

int transCharToInt(char *_buffer,int _start,int _stop)  //char to int
{
  int _index;
  int result = 0;
  int num = _stop - _start + 1;
  int _temp[num];
  for (_index = _start;_index <= _stop;_index ++)
  {
    _temp[_index - _start] = _buffer[_index] - '0';
    result = 10*result + _temp[_index - _start];
  }
  return result;
}

int WindDirection()   //Wind Direction
{
  return transCharToInt(databuffer,1,3);
}

// 1 rev/sec = 1.492mph = 2.4km/h = 0.67m/s
float WindSpeedAverage() //air Speed (1 minute)
{
  temp = 0.66698368 * transCharToInt(databuffer,5,7);
  return temp;
}

float WindSpeedMax() //Max air speed (5 minutes)
{
  temp = 0.66698368 * transCharToInt(databuffer,9,11) ;
  return temp;
}

float Temperature() //Temperature ("C")
{
  temp = (transCharToInt(databuffer,13,15) - 32.00) * 5.00 / 9.00;
  return temp ;
}


float RainfallOneHour() //Rainfall (1 hour)
{
  temp = transCharToInt(databuffer,17,19) * 27.94 * 0.01;
  return temp;
}

float RainfallOneDay() //Rainfall (24 hours)
{
  temp = transCharToInt(databuffer,21,23) * 27.94 * 0.01;
  return temp;
}

int Humidity() //Humidity
{
  return transCharToInt(databuffer,25,26);
}

float BarPressure() //Barometric Pressure
{
  temp = transCharToInt(databuffer,28,32);
  return temp / 10.00;
}
```